

THIS PAGE IS INSERTED BY OIPE SCANNING

IMAGES WITHIN THIS DOCUMENT ARE BEST AVAILABLE COPY AND CONTAIN DEFECTIVE IMAGES SCANNED FROM ORIGINALS SUBMITTED BY THE APPLICANT.

DEFECTIVE IMAGES COULD INCLUDE BUT ARE NOT LIMITED TO:

BLACK BORDERS

TEXT CUT OFF AT TOP, BOTTOM OR SIDES

FADED TEXT

ILLEGIBLE TEXT

SKEWED/SLANTED IMAGES

COLORLED PHOTOS

BLACK OR VERY BLACK AND WHITE DARK PHOTOS

GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.
RESCANNING DOCUMENTS *WILL NOT*
CORRECT IMAGES.**

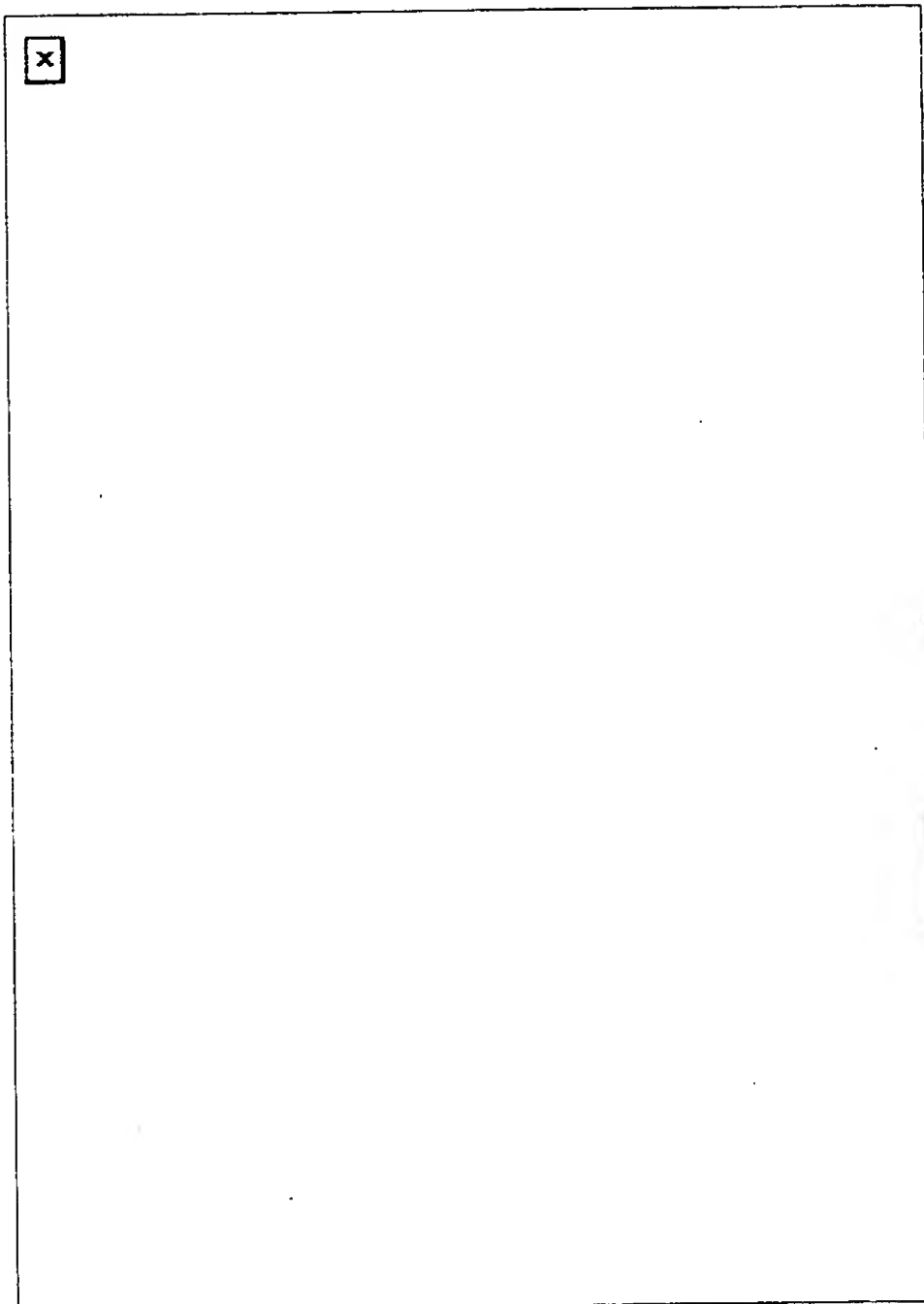
STORAGE DEVICE

Patent number: JP7210439
Publication date: 1995-08-11
Inventor: KANDA MOTOHIRO; others: 02
Applicant: HITACHI LTD
Classification:
- international: G06F12/00; G06F3/06
- european:
Application number: JP19940290947 19941125
Priority number(s):

Abstract of JP7210439

PURPOSE: To provide a storage device capable of executing dump processing without interposing CPU and completely dumping the data to be dumped at the time of starting dump even in the case of executing data update between the start of dump and the finish of it.

CONSTITUTION: A disk device 30 with a dumping function transfers data in the data unupdate area of a magnetic disk 100 to a disk device 31 at a dumping destination in a prescribed order. At the time of receiving data update requirement between the start of dump and the finish of it, the data to be written is written into the magnetic disk 100 when it is in an area excepting for the data unupdate area, and a data block before update is retreated to a buffer 70 and then the data to be written is written into the magnetic disk 100 when the data to be written is in the data unupdate area. After then, at the point of the time earliest of the times when data quantity which is not retreated to the buffer 70 becomes over a specified quantity and when transferring pertinent data to the disk device 31 when the update of pertinent data is not required, data retreated to the buffer 70 is transferred to the disk device 31 at a dump destination.



Data supplied from the *esp@cenet* database - Worldwide

W1383

(19)日本国特許庁 (J P) (12) 公開特許公報 (A) (11)特許出願公開番号
特開平7-210439
(43)公開日 平成7年(1995)8月11日

(51)Int.Cl.⁶ 識別記号 庁内整理番号 F I 技術表示箇所
G 0 6 F 12/00 5 3 1 M 8944-5B
3/06 3 0 1 X

審査請求 未請求 請求項の数18 O L (全 24 頁)

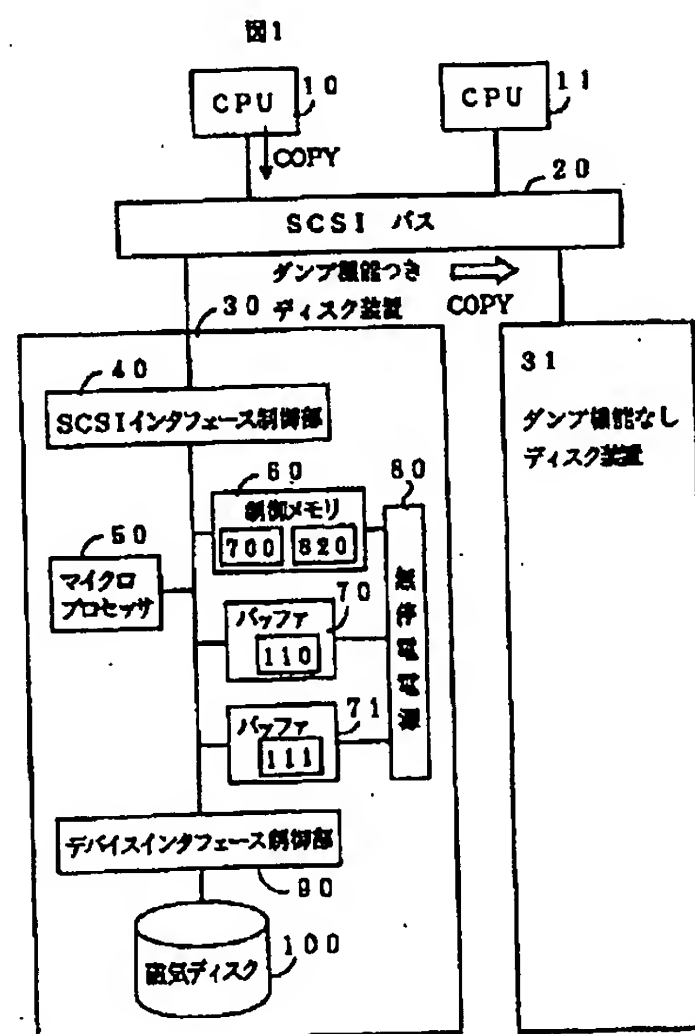
(21)出願番号	特願平6-290947	(71)出願人	000005108 株式会社日立製作所 東京都千代田区神田駿河台四丁目6番地
(22)出願日	平成6年(1994)11月25日	(72)発明者	神田 基博 神奈川県川崎市麻生区王禅寺1099番地 株 式会社日立製作所システム開発研究所内
(31)優先権主張番号	特願平5-303669	(72)発明者	佐藤 孝夫 神奈川県小田原市国府津2880番地 株式会 社日立製作所ストレージシステム事業部内
(32)優先日	平5(1993)12月3日	(72)発明者	山本 彰 神奈川県川崎市麻生区王禅寺1099番地 株 式会社日立製作所システム開発研究所内
(33)優先権主張国	日本 (J P)	(74)代理人	弁理士 小川 勝男

(54)【発明の名称】 記憶装置

(57)【要約】

【目的】CPUを介在させずにダンプ処理を行うことが出来ると共に、ダンプ開始からダンプ終了までの間にデータ更新を行なっても、ダンプ開始時点でのダンプ対象データを完全にダンプすることが出来る記憶装置を提供する。

【構成】ダンプ機能つきディスク装置30は、磁気ディスク100のデータ未更新領域のデータを所定の順にダンプ先のディスク装置31へ転送する。ダンプ開始からダンプ終了までの間にデータ更新要求を受けると、ライト対象データの領域がデータ未更新領域以外であればライト対象データを磁気ディスク100に書き込み、前記データ未更新領域であれば更新前データブロックをバッファ70に退避し、その後でライト対象データを磁気ディスク100に書き込む。その後、バッファ70に退避したデータ量が所定量を越えた契機か又は当該データに対する更新要求がなかった場合に当該データをダンプ先のディスク装置31へ転送する契機のいずれか早い方に、バッファ70に退避したデータをダンプ先のディスク装置31へ転送する。



【特許請求の範囲】

【請求項1】記憶媒体に格納されているデータを別の記憶装置へダンプする機能を有する記憶装置において、記憶媒体のダンプ対象領域を、ダンプを行ったダンプ済み領域とダンプを行っていない未ダンプ領域とに分けて認識し、且つ、前記未ダンプ領域を、データを更新したデータ更新済み領域とデータを更新していないデータ未更新領域とに分けて認識する領域認識手段と、ダンプ開始からダンプ終了までの間に、前記データ未更新領域以外のデータに対するデータ更新要求を受けたときは当該データを更新し、一方、前記データ未更新領域のデータに対するデータ更新要求を受けたときは当該データをバッファに退避し、その後で前記データを更新し、当該データの記憶領域をデータ未更新領域からデータ更新済み領域に変更するデータ更新手段と、記憶媒体のデータ未更新領域のデータを、所定の順にダンプ先の記憶装置へ転送し、転送したデータの記憶領域を未ダンプ領域からダンプ済み領域に変更するデータ転送手段と、前記バッファに退避したデータを所定の契機でダンプ先の記憶装置へ転送すると共に、当該データの記憶領域を未ダンプ領域からダンプ済み領域に変更し、当該データをバッファから実質的に消去する退避データ転送手段とを具備したことを特徴とする記憶装置。

【請求項2】請求項1記載の記憶装置において、バッファに退避したデータをダンプ先の記憶装置へ転送する契機が、バッファに退避したデータ量が所定量を越えた契機か又は当該データに対する更新要求がなかった場合に当該データをダンプ先の記憶装置へ転送する契機のいずれか早い方であることを特徴とする記憶装置。

【請求項3】請求項1記載の記憶装置において、バッファに退避したデータをダンプ先の記憶装置へ転送する契機が、バッファに退避したデータ量のバッファ容量に対する割合が所定値を越えた契機であることを特徴とする記憶装置。

【請求項4】請求項1記載の記憶装置において、バッファに退避したデータをダンプ先の記憶装置へ転送する契機が、当該データに対する更新要求がなかった場合に当該データをダンプ先の記憶装置へ転送する契機であることを特徴とする記憶装置。

【請求項5】請求項1記載の記憶装置において、バッファに退避したデータをダンプ先の記憶装置へ転送する契機が、当該データをバッファに退避した直後から当該データに対する更新要求の完了直前までの間の契機であることを特徴とする記憶装置。

【請求項6】請求項1記載の記憶装置において、バッファに退避したデータをダンプ先の記憶装置へ転送する契機が、所定の時間ごとであることを特徴とする記憶装置。

【請求項7】請求項1から請求項6のいずれかに記載の

記憶装置において、バッファを多重化したことを特徴とする記憶装置。

【請求項8】請求項1から請求項7のいずれかに記載の記憶装置において、バッファを不揮発性としたことを特徴とする記憶装置。

【請求項9】記憶媒体に格納されているデータを別の記憶装置へダンプする機能を有する記憶装置において、記憶媒体のダンプ対象領域を、ダンプを行ったダンプ済み領域とダンプを行っていない未ダンプ領域とに分けて認識する領域認識手段と、

記憶媒体の未ダンプ領域のデータを所定の順にダンプ先の記憶装置へ転送し、転送したデータの記憶領域を未ダンプ領域からダンプ済み領域に変更すると共に、ダンプ開始からダンプ終了までの間に未ダンプ領域のデータに対するデータ更新要求を受けたときは当該データをダンプ先の記憶装置へ優先的に転送し、当該データの記憶領域を未ダンプ領域からダンプ済み領域に変更するデータ転送手段と、

ダンプ開始からダンプ終了までの間に、前記未ダンプ領域以外のデータに対するデータ更新要求を受けたときは当該データを更新し、一方、前記未ダンプ領域のデータに対するデータ更新要求を受けたときは前記優先的転送の後で前記データを更新するデータ更新手段とを具備したことを特徴とする記憶装置。

【請求項10】請求項1から請求項9のいずれかに記載の記憶装置において、ダンプ対象データを転送するダンプ先の記憶装置における記憶領域を、当該データに対する更新要求がなかった場合にデータ未更新領域の当該データを転送する記憶領域と等しくすることを特徴とする記憶装置。

【請求項11】請求項1から請求項8のいずれかに記載の記憶装置において、ダンプ対象データを転送するダンプ先の記憶装置における記憶領域を、当該データの転送の直前に他のデータが転送されていた記憶領域の次の記憶領域とすることを特徴とする記憶装置。

【請求項12】請求項11に記載の記憶装置において、ダンプ対象データに対してそのデータの元の記憶媒体上の記憶領域を指すLBA(Logical Block Address)を含む識別子を付加してダンプ先の記憶装置に転送するデータとする識別子付加手段を具備したことを特徴とする記憶装置。

【請求項13】請求項1から請求項8のいずれかに記載の記憶装置において、ダンプ先の記憶装置において、データ未更新領域のデータを格納する記憶媒体と、バッファに待避したデータを格納する記憶媒体とが異なることを特徴とする記憶装置。

【請求項14】請求項9に記載の記憶装置において、ダンプ先の記憶装置において、未ダンプ領域のデータを格納する記憶媒体と、データ更新要求を受けたデータを格納する記憶媒体とが異なることを特徴とする記憶装置。

【請求項15】請求項13から請求項14のいずれかに記載の記憶装置において、バッファに待避したデータあるいはデータ更新要求を受けたデータに対してそのデータの元の記憶媒体上の記憶領域を指すLBA(Logical Block Address)を含む識別子を付加してダンプ先の記憶装置に転送するデータとする識別子付加手段を具備したことを特徴とする記憶装置。

【請求項16】請求項12および請求項15のいずれかに記載の記憶装置において、前記識別子が、当該データの「更新時刻」や「ダンプ時刻」などの時刻情報を含むことを特徴とする記憶装置。

【請求項17】請求項1から請求項16のいずれかに記載の記憶装置において、ダンプ先の記憶装置に記憶されたデータから、ダンプ開始時点でダンプ元の記憶媒体に記憶されていたデータを復元するデータ復元手段を具備したことを特徴とする記憶装置。

【請求項18】請求項1から請求項17のいずれかに記載の記憶装置において、SCSI規格の記憶装置であることを特徴とする記憶装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、記憶装置に関し、さらに詳しくは、CPUを介在させずにダンプ処理を行うことが出来ると共に、ダンプ開始からダンプ終了までの間にデータ更新を行なっても、ダンプ開始時点でのダンプ対象データを完全にダンプできる記憶装置に関する。

【0002】

【従来の技術】「最新SCSIマニュアル 1989年 CQ出版社 177頁」には、SCSI規格で定義された「COPYコマンド」を用いることで、CPU(SCSI規格ではイニシエータと呼ばれる)が介入することなく、一つの記憶装置から別の記憶装置へダンプ処理を行う機能が記載されている。

【0003】また、特開平5-210555号公報には、記憶媒体に格納されているデータを別の記憶装置へダンプする機能を有する記憶装置が開示されている。この記憶装置では、ダンプ対象のデータを所定の順序でCPUにコピーさせ、そのコピーしたデータをCPUが別の記憶装置に転送する。ダンプ処理と並行して行った他のジョブにより、ダンプ対象でありながら未だダンプを行っていないデータの更新要求が発生した場合、更新前データをサイドファイルに退避した後、当該データの更新を実行する。その後、サイドファイル内に退避したデータをCPUにコピーさせる。CPUは、当該データに対する更新要求がなかった場合に当該データをダンプ先の記憶装置へ転送する契機と同じ契機で、当該データをダンプ先の記憶装置へ転送する。

【0004】他の関連する従来技術は、例えば特開昭57-90770号公報や、特開平1-231150号公報に開示されている。

【0005】

【発明が解決しようとする課題】上記SCSI規格で定義された「COPYコマンド」を用いてダンプ処理を行う従来技術では、CPUに負担をかけずにダンプ処理を行うことが出来る。しかし、ダンプ開始からダンプ終了までの間に、記憶媒体のダンプ対象領域中の未だダンプを行っていない領域のデータ更新を行なうと、更新前のデータが失われてしまう。このため、ダンプ開始からダンプ終了までデータ更新を行なえず、ダンプ対象の記憶装置を他のジョブの処理に使えない問題点がある。

【0006】一方、上記特開平5-210555号公報に開示された従来の記憶装置では、ダンプ開始からダンプ終了までの間にデータ更新を行なっても、更新前のデータが失われないから、ダンプ対象の記憶装置を他のジョブの処理に使うことが出来る。しかし、CPUを介在させてダンプ処理を行うため、CPUに負担をかける問題点がある。

【0007】そこで、本発明の目的は、CPUを介在させずにダンプ処理を行うことが出来ると共に、ダンプ開始からダンプ終了までの間にデータ更新を行なっても、ダンプ開始時点でのダンプ対象データを完全にダンプできる記憶装置を提供することにある。

【0008】

【課題を解決するための手段】第1の観点では、本発明は、記憶媒体に格納されているデータを別の記憶装置へダンプする機能を有する記憶装置において、記憶媒体のダンプ対象領域を、ダンプを行ったダンプ済み領域とダンプを行っていない未ダンプ領域とに分けて認識し、且つ、前記未ダンプ領域を、データを更新したデータ更新済み領域とデータを更新していないデータ未更新領域とに分けて認識する領域認識手段と、ダンプ開始からダンプ終了までの間に、前記データ未更新領域以外のデータに対するデータ更新要求を受けたときは当該データを更新し、一方、前記データ未更新領域のデータに対するデータ更新要求を受けたときは当該データをバッファに退避し、その後で前記データを更新し、当該データの記憶領域をデータ未更新領域からデータ更新済み領域に変更するデータ更新手段と、記憶媒体のデータ未更新領域のデータを、所定の順にダンプ先の記憶装置へ転送し、転送したデータの記憶領域を未ダンプ領域からダンプ済み領域に変更するデータ転送手段と、前記バッファに退避したデータを所定の契機でダンプ先の記憶装置へ転送すると共に、当該データの記憶領域を未ダンプ領域からダンプ済み領域に変更し、当該データをバッファから実質的に消去する退避データ転送手段とを具備したことを特徴とする記憶装置を提供する。

【0009】第2の観点では、本発明は、記憶媒体に格納されているデータを別の記憶装置へダンプする機能を有する記憶装置において、記憶媒体のダンプ対象領域を、ダンプを行ったダンプ済み領域とダンプを行ってい

ない未ダンプ領域とに分けて認識する領域認識手段と、記憶媒体の未ダンプ領域のデータを所定の順にダンプ先の記憶装置へ転送し、転送したデータの記憶領域を未ダンプ領域からダンプ済み領域に変更すると共に、ダンプ開始からダンプ終了までの間に未ダンプ領域のデータに対するデータ更新要求を受けたときは当該データをダンプ先の記憶装置へ優先的に転送し、当該データの記憶領域を未ダンプ領域からダンプ済み領域に変更するデータ転送手段と、ダンプ開始からダンプ終了までの間に、前記未ダンプ領域以外のデータに対するデータ更新要求を受けたときは当該データを更新し、一方、前記未ダンプ領域のデータに対するデータ更新要求を受けたときは前記優先的転送の後で前記データを更新するデータ更新手段とを具備したことを特徴とする記憶装置を提供する。

【0010】

【作用】上記第1の観点による記憶装置では、ダンプを行っていない未ダンプ領域中の、データを更新していないデータ未更新領域を、領域認識手段により認識し、そのデータ未更新領域のデータに対する更新要求があったときは、データ更新手段により更新前データをバッファに退避してから更新を行うようにした。そして、ダンプ元の記憶媒体のデータはデータ転送手段によりダンプ先の記憶装置に転送し、バッファに退避したデータは退避データ転送手段によりダンプ先の記憶装置に転送するようにした。

【0011】このため、CPUを介在させずにダンプ処理を行うことが出来ると共に、ダンプ開始からダンプ終了までの間にデータ更新を行なっても、ダンプ開始時点でのダンプ対象データを完全にダンプすることが出来る。

【0012】なお、バッファに待避した更新前データをダンプ先の記憶装置に転送する契機は、バッファの容量、ダンプ処理に要求される信頼性、あるいは、ダンプ処理と並行して行われる更新処理の応答時間性能などを考慮して、最適なものに定めればよい。

【0013】また、バッファを多重化すれば、いずれかのバッファが故障してもデータ消失を回避できる。さらに、バッファを不揮発性とすれば、主電源に障害が発生してもデータ消失を回避でき、ダンプ処理の信頼性を向上できる。

【0014】上記第2の観点による記憶装置では、領域認識手段により、ダンプを行っていない未ダンプ領域を認識し、データ転送手段により、前記未ダンプ領域のデータを所定の順にダンプ先の記憶装置へ転送し、ダンプ開始からダンプ終了までの間に未ダンプ領域のデータに対するデータ更新要求を受けたときは当該データをダンプ先の記憶装置へ優先的に転送し、その後、データ更新手段により、データを更新するようにした。

【0015】このため、CPUを介在させずにダンプ処

理を行うことが出来ると共に、ダンプ開始からダンプ終了までの間にデータ更新を行なっても、ダンプ開始時点でのダンプ対象データを完全にダンプすることが出来る。さらに、やや応答時間は長くなるが、バッファを不要とすることも出来る。

【0016】なお、ダンプ対象データを転送するダンプ先の記憶装置における記憶領域を、当該データに対する更新要求がなかった場合にデータ未更新領域の当該データを転送する記憶領域と等しくすれば、ダンプ先の記憶装置がランダムアクセスに適したものである場合に好適となる。

【0017】一方、ダンプ対象データを転送するダンプ先の記憶装置における記憶領域を、当該データの転送の直前に他のデータが転送されていた記憶領域の次の記憶領域とすれば、ダンプ先の記憶媒体がシーケンシャルアクセスに適したものである場合に好適となる。このとき、ダンプ対象データに対してそのデータの元の記憶媒体上の記憶領域を指すLBA(Logical Block Address)を含む識別子を付加してダンプ先の記憶装置に転送すれば、当該データの位置の復元も可能となる。さらに、前記識別子に時刻情報を含めれば、更新前データと更新後データとを重複してダンプしても識別可能となる。

【0018】また、ダンプ先の記憶装置において、データ更新要求により待避する必要の生じた更新前データを格納する記憶媒体と、それ以外の通常のダンプ処理によるデータを格納する記憶媒体とを別にすれば、通常のダンプ処理によるデータの格納が、更新前データの格納により中断されることなく、ダンプ処理を高速に行うことができる。さらに、前記記憶媒体が、シーケンシャルアクセスに適したものである場合には、更新前データにはLBAを含む識別子を付加して格納する必要がある一方、通常のダンプ処理によるデータは識別子を付加することなく、ダンプ元の記憶媒体上の記憶領域の位置の順に格納しても、データの復元が可能である。これは、前記2種類のデータを、別々の記憶媒体に格納するためである。

【0019】

【実施例】以下、図に示す実施例によりこの発明をさらに詳しく説明する。なお、これによりこの発明が限定されるものではない。

【0020】(実施例1)図1は、本発明の第1実施例の記憶装置を含む計算機システムの要部構成図である。CPU10とCPU11は、SCSIバス20を介して、本発明の第1実施例の記憶装置であるダンプ機能つきディスク装置30と、公知のダンプ機能なしディスク装置31とに接続されている。ダンプ機能つきディスク装置30およびダンプ機能なしディスク装置31は、どちらもブロック長512バイト、容量100MBを持っている。

【0021】なお、説明の都合上、ダンプ機能つきディ

スク装置30のデータをデータブロックごとにダンプ機能なしディスク装置31へダンプするものとする。ただし、これに限定されず、例えば複数ブロックずつダンプしてもよい。

【0022】ダンプ機能つきディスク装置30は、SCSIインタフェース制御部40と、マイクロプロセッサ50と、制御メモリ60と、2重化されたバッファ70、71と、無停電電源80と、デバイスインタフェース制御部90と、磁気ディスク100とを有している。また、ダンプ機能つきディスク装置30は、SCSIで定義された「COPYコマンド」処理中であっても、新たなコマンドの要求があった場合に、それをデバイスビジステータスでエラー終了させることなく、正常に処理する機能を有している。

【0023】SCSIインタフェース制御部40は、ダンプ機能つきディスク装置30とSCSIバス20との間で、各種の信号を授受する。マイクロプロセッサ50は、各部の作動を制御する。

【0024】制御メモリ60は、マイクロプロセッサ50のプログラムや制御データを格納する半導体メモリであり、更新前データ情報保持テーブル700と、コピーポインタ820とを備える。更新前データ情報保持テーブル700については、図2を参照して後で説明する。コピーポインタ820は、ダンプを行っていない未ダンプ領域の先頭ブロックのLBA(Logical Block Address)を指すポインタである。

【0025】バッファ70、71は、データを一時的に格納する半導体メモリであり、容量5120バイトの更新前データ保持用領域110、111を有し、それらに最大10個の更新前データブロックを2重化して保持できる。

【0026】無停電電源80は、バッテリーを備え、ディスク装置30の主電源(図示せず)が故障しても、制御メモリ60、バッファ70、71への電力供給を維持し、半導体メモリを不揮発性メモリ化する。デバイスインタフェース制御部90は、磁気ディスク100との間で読出データや書込データの授受を行なう。

【0027】図2は、更新前データ情報保持テーブル700の構成図である。更新前データ情報保持テーブル700は、更新前データ保持用領域110、111(図1)に保持された更新前データブロックごとに対応するエントリを持つ。エントリは、エントリ状態フィールド700aと、LBAフィールド700bと、更新前データブロックアドレスフィールド700cと、更新前データブロックアドレスフィールド700dとを持つ。

【0028】エントリ状態フィールド700aには、“未使用”、“未ダンプ”、“ダンプ済み”のいずれか1つを格納する。“未使用”は、当該エントリが使用されていないことを示す。“未ダンプ”は、未だダンプしていない更新前データブロックが更新前データ保持用領

域110、111に保持されていることを示す。“ダンプ済み”は、更新前データブロックが、既にダンプした更新前データブロックが更新前データ保持用領域110、111に保持されていることを示す。

【0029】LBAフィールド700bには、更新前データブロックの磁気ディスク100における記憶領域を指すLBAを格納する。

【0030】更新前データブロックアドレスフィールド700cには、更新前データブロックが保持されている更新前データ保持用領域110におけるアドレスを格納する。更新前データブロックアドレスフィールド700dには、更新前データブロックが保持されている更新前データ保持用領域111におけるアドレスを格納する。

【0031】図3は、ダンプ処理の手順を説明するフローチャートである。ステップ210では、CPU10は、SCSI規格で定義された「RESERVEコマンド」により、ダンプ先であるダンプ機能なしディスク装置31を排他制御し、他のCPU11からのアクセスを禁止する。ただし、ダンプ元であるダンプ機能つきディスク装置30に対しては、排他制御をしない。

【0032】ステップ220では、CPU10は、ダンプ機能つきディスク装置30に、「COPYコマンド」を発行する。この「COPYコマンド」では、ブロックデバイスからブロックデバイスへのダンプであることと、ダンプ機能つきディスク装置30およびダンプ機能なしディスク装置31のSCSI_idと、LU(Logical Unit)番号と、コピー開始LBAと、ダンプするブロック数とが指定される。数値例として、ダンプ機能つきディスク装置30およびダンプ機能なしディスク装置31におけるコピー開始LBAはいずれも“0”とし、ダンプするブロック数は100MB/512バイト=204,800個とする。

【0033】ステップ230では、ダンプ機能つきディスク装置30は、CPU10から「COPYコマンド」を受けとり、SCSI規格で定義された「DISCONNECTコマンド」をCPU10に発行し、SCSIバス20との接続を切断する。これは、ダンプ処理に長時間を要するため、CPU10とSCSIバス20を占有しないようにするためである。

【0034】ステップ235では、ダンプ機能つきディスク装置30は、図4に示すCOPYコマンド処理を実行する。

【0035】図4は、COPYコマンド処理のフローチャートである。ステップ320では、「COPYコマンド」で指定されたコピー開始LBAを、制御メモリ60のコピーポインタ820(図1)に設定する。また、更新前データ情報保持テーブル700(図2)のすべてのエントリのエントリ状態フィールド700aを“未使用”に設定する。

【0036】ステップ330では、更新前データ情報保持テーブル700の全てのエントリを調べ、エントリ状態フィールド700aが“ダンプ済み”または“未ダンプ”であり且つコピーポインタ820の指すLBAをLBAフィールド700bに持つエントリを検索する。前記エントリが見つからなければステップ340に進み、見つければステップ345に進む。なお、更新前データ情報保持テーブル700にハッシュやビットマップやインデクスなどのデータ構造を採用すれば、上記検索を高速に行うことが出来る。

【0037】ステップ340では、コピーポインタ820が指すLBAに対応したデータブロックを磁気ディスク100から読み出し、SCSI規格で定義された「WRITEコマンド」によりダンプ機能なしディスク装置31へ転送する。この「WRITEコマンド」では、ダンプ機能なしディスク装置31のLU番号と、転送するデータブロックのLBAと、転送長“1ブロック”とを指定する。そして、ステップ380に進む。

【0038】ステップ345では、上記ステップ330で見つかったエントリのエントリ状態フィールド700aを調べ、“未ダンプ”であればステップ350に進み、“ダンプ済み”であればステップ370に進む。

【0039】ステップ350では、当該“未ダンプ”のエントリが指すバッファ70の更新前データ保持用領域110の更新前データブロック（またはバッファ71の更新前データ保持用領域111の更新前データブロックのいずれか）を読み出し、「WRITEコマンド」によりダンプ機能なしディスク装置31へ転送する。この「WRITEコマンド」では、ダンプ機能なしディスク装置31のLU番号と、ダンプ先のLBAと、転送長“1ブロック”とを指定する。ダンプ機能なしディスク装置31のようにダンプ先がランダムアクセスに適した記憶媒体である場合には、ダンプ先のLBAは、磁気ディスク100で当該更新前データブロックが書き込まれていたLBAと一致させる。これにより、ダンプ元とダンプ先とで、更新前データブロックの書き込み領域（位置）を一致させることが出来る。

【0040】ステップ360では、当該更新前データブロックを、更新前データ保持用領域110と更新前データ保持用領域111の両方から消去する。ステップ370では、前記ステップ335における“ダンプ済み”のエントリまたはステップ360において消去した更新前データブロックを指すエントリのエントリ状態フィールド700aを“未使用”に変更する。

【0041】ステップ380では、制御メモリ60のコピーポインタ820を、次のデータブロックのLBAを指すように設定する。ステップ390では、コピーポインタ820とコピー終了LBA（＝コピー開始LBA＋ダンプするブロック数）とを比較する。コピーポインタ820≧コピー終了LBAでないならば、前記ステップ

330に戻る。コピーポインタ820≧コピー終了LBAならば、COPYコマンド処理を終了する。

【0042】図3に戻り、ステップ240、ステップ250が、上記COPYコマンド処理と並行して実行される。ステップ240では、例えばCPU11は、ダンプ機能つきディスク装置30に「WRITEコマンド」を発行し、磁気ディスク100のデータブロックを更新しようとする。この「WRITEコマンド」では、ダンプ機能つきディスク装置30のLU番号と、ライト対象データのLBAと、ライトするブロック数とが指定される。

【0043】ステップ250では、ダンプ機能つきディスク装置30は、図5に示すWRITEコマンド処理を実行する。

【0044】図5は、WRITEコマンド処理のフローチャートである。ステップ510では、CPU11からWRITEコマンドを受け取る。ステップ515では、COPYコマンド処理中でなければステップ516に進み、COPYコマンド処理中であればステップ520に進む。

【0045】ステップ516では、「WRITEコマンド」で指定されたライト対象データを、デバイスインタフェース制御部90を経由して、磁気ディスク100に書き込む。これは従来の書き込み方法と同じである。

【0046】ステップ518では、CPU11に、ステータスバイトとコマンドコンプリートメッセージを送り、WRITEコマンドの完了を報告して、WRITEコマンド処理を終了する。

【0047】ステップ520では、コピーポインタ820の示すLBA≦ライト対象データのLBA≦コピー終了LBAであるなら（これは、ライト対象データが未ダンプ領域のデータであることを意味する）、ステップ525に進む。コピーポインタ820の示すLBA≦ライト対象データのLBA≦コピー終了LBAでないなら（これは、ライト対象データが未ダンプ領域のデータでないことを意味する）、ステップ541に進む。

【0048】ステップ525では、更新前データ情報保持テーブル700の全てのエントリを調べ、エントリ状態フィールド700aが“ダンプ済み”または“未ダンプ”であり且つライト対象データのLBAを含むLBAをLBAフィールド700bに持つエントリを検索する。前記エントリが見つからなければ（これは、ライト対象データがデータ未更新領域のデータであることを意味する）、ステップ530に進む。見つければ（これは、ライト対象データがデータ更新済領域のデータであることを意味する）、ステップ341に進む。

【0049】ステップ530では、ライト対象データの更新前データを含む更新前データブロックを、バッファ70、71の更新前データ保持用領域110、111に読み込む。

【0050】ステップ540では、当該更新前データブロックに対応するエントリを、更新前データ情報保持テーブル700に追加登録する。具体的には、新たなエントリを確保し、そのエントリのエントリ状態フィールド700aに“未ダンプ”を格納し、LBAフィールド700bに当該更新前データブロックのLBAを格納し、更新前データブロックアドレスフィールド700c、700dに当該更新前データブロックの更新前データ保持用領域110、111におけるアドレスをそれぞれ格納する。

【0051】ステップ541では、当該ライト対象データを磁気ディスク100に書き込む。ステップ543では、「WRITEコマンド」で指定された全てのライト対象データを磁気ディスク100に書き込んだか否か判定する。全てのライト対象データを書き込んでいればステップ560に進み、書き込んでいないライト対象データがあれば前記ステップ520に戻る。

【0052】ステップ560では、CPU11に、ステータスバイトとコマンドコンプリートメッセージを送り、WRITEコマンド処理の完了を報告する。

【0053】ステップ570では、更新前データ保持用領域110、111が更新前データブロックで“フル”になったか否か判定する。“フル”になっていればステップ580に進み、“フル”でなければWRITEコマンド処理を終了する。具体的には、更新前データ情報保持テーブル700の各エントリを調べ、エントリ状態フィールド700aに“未ダンプ”が格納されているエントリの数が“10”(=更新前データ保持用領域110の最大のブロック数)に等しければ“フル”と判定し、“10”未満であれば“フル”でないと判定する。

【0054】ステップ580では、ダンプ機能つきディスク装置30は、バッファ70(またはバッファ71のいずれでもよい)に保持された更新前データブロックをダンプ機能なしディスク装置31に転送し、バッファ70、71を空にするフラッシュ処理を行なう。CPU11へコマンドコンプリートメッセージを送信した後(前記ステップ560)で、このフラッシュ処理を行なうのは、フラッシュ処理に長時間を要するので、その完了までCPU11を占有しないためである。

【0055】図6は、フラッシュ処理のフローチャートである。ステップ600では、更新前データ情報保持テーブル700の最初のエントリを処理対象のエントリとして指定する。

【0056】ステップ605では、処理対象のエントリのエントリ状態フィールド700aが“未ダンプ”であればステップ610に進み、“未使用”または“ダンプ済み”であればステップ640に進む。

【0057】ステップ610では、処理対象のエントリの更新前データブロックアドレスフィールド700c(または700d)が指すバッファ70の更新前データ

保持用領域110の更新前データブロック(またはバッファ71の更新前データ保持用領域111の更新前データブロック)を読み出し、先述のステップ350と同様にして「WRITEコマンド」によりダンプ機能なしディスク装置31へ転送する。

【0058】ステップ620では、転送した更新前データブロックをバッファ70、71の更新前データ保持用領域110、111の両方から消去する。ステップ630では、更新前データ情報保持テーブル700の当該更新前データブロック対応のエントリのエントリ状態フィールド700aを“ダンプ済み”に変更する。

【0059】ステップ640では、処理対象のエントリが更新前データ情報保持テーブル700の最後のエントリであればフラッシュ処理を終了し、最後のエントリでなければステップ650に進む。ステップ650では、処理対象のエントリとして次のエントリを指定して、前記ステップ605に戻る。

【0060】図3に戻り、ステップ260では、ダンプ機能つきディスク装置30は、SCSI規格で定義された「RESELECT」をCPU10に発行し、SCSIバス20に再び接続する。ステップ270では、ダンプ機能つきディスク装置30は、CPU10にステータスバイトとコマンドコンプリートメッセージを送り、COPYコマンド処理の終了を通知する。

【0061】ステップ280では、CPU10は、COPYコマンド処理の終了を受領すると、ダンプ機能なしディスク装置31に「RELEASEコマンド」を発行して、排他制御を解除する。

【0062】図7は、以上の動作を模式的に示す概念図である。ダンプ機能つきディスク装置30は、制御メモリ60のコピーポインタ820を、ダンプ元の磁気ディスク100のダンプ対象領域のダンプ開始LBAかダンプ終了LBAまで1ブロックずつ順に移動させながら、ポイントされたデータブロックをダンプ先の磁気ディスク800の同じLBAの領域に転送する。

【0063】ダンプ開始からダンプ終了までの間(COPYコマンド処理中)に、「WRITEコマンド」を受けると、ライト対象データに対応する更新前データブロック850を更新前データ保持用領域110(および111)の領域851に退避する。

【0064】更新前データブロック850が書き込まれていた領域P1までコピーポインタ820が来た時か又は更新前データ保持用領域110が“フル”になった時のいずれか早い方の契機に、更新前データ保持用領域110(または111)の領域851に退避していた更新前データブロック850をダンプ先の磁気ディスク800の領域852に転送する。

【0065】上記第1実施例のダンプ機能つきディスク装置30によれば、CPUを介在させずにダンプ処理を行うことが出来ると共に、ダンプ開始からダンプ終了ま

での間にデータ更新を行なっても、ダンプ開始時点でのダンプ対象データを完全にダンプすることが出来る。

【0066】また、バッファ70、71によりバッファを2重化しているため、一方が故障してもデータ消失を回避でき、さらに、無停電電源80によりバッファ70、71を不揮発性メモリ化しているため、主電源に障害が発生してもデータ消失を回避でき、ダンプ処理の信頼性を向上できる。

【0067】上記第1実施例は、次のように変形してもよい。

①上記第1実施例では、バッファに退避したデータをダンプ先の記憶装置へ転送する契機が、バッファが“フル”になった契機か又は当該データに対する更新要求がなかった場合に当該データをダンプ先の記憶装置へ転送する契機のいずれか早い方であったが、単純に、バッファ使用量が所定量（例えばバッファ容量の80%）を越えた契機としてもよい。この場合、COPYコマンド処理（図4）で、コピーポインタ820の示すLBAが更新前データ情報保持テーブル700にあったとき（ステップ330）でも、ダンプ先への書き込みを行わずにステップ380に進み、バッファ使用量が所定量を越えた契機でフラッシュ処理（図6）を行えばよい。

【0068】②上記第1実施例では、更新前データ保持用領域110、111の記憶容量を10データブロック分としたが、バッファを充分大きな記憶容量のものとするれば、バッファに退避したデータをダンプ先の記憶装置へ転送する契機を、単純に、所定時間ごと（例えば30秒）としたり、当該データに対する更新要求がなかった場合に当該データをダンプ先の記憶装置へ転送する契機としたり、当該データをバッファに退避した直後から当該データに対する更新要求の完了直前までの間の契機とすることが出来る。

【0069】③上記第1実施例では、更新前データ保持用領域110、111の記憶容量を固定としたが、更新前データブロックの格納量に応じて記憶容量を増大させるようにしてもよい。この場合、バッファに退避したデータをダンプ先の記憶装置へ転送する契機は、上記②と同じに出来る。

【0070】④上記第1実施例では、WRITEコマンド処理（図5）の中でフラッシュ処理を行なったが、WRITEコマンド処理やCOPYコマンド処理とは非同期に独立して行ってもよい。

【0071】⑤上記第1実施例では、ダンプ先の記憶装置としてディスク装置31を使用したか、ディスク装置31以外の記憶媒体を使用してもよい。

⑥上記第1実施例では、SCSIインタフェース40を備えたが、その代りに又はそれに加えて、SCSI以外のインタフェースを備えてもよい。

【0072】⑦上記第1実施例では、ダンプ元とダンプ先とを別々のディスク装置としたが、ダンプ機能つきデ

ィスク装置30（図1）に、ダンプ先の記憶媒体を設けることも出来る。例えば、デバイスインタフェース部90に、磁気ディスク100に加えて、ダンプ先の記憶媒体を接続してもよい。

【0073】（実施例2）図8は、本発明の第2実施例による記憶装置を含む計算機システムの要部構成図である。

【0074】本発明の第2実施例の記憶装置であるダンプ機能つきディスク装置30'の構成は、基本的に、第1実施例のダンプ機能つきディスク装置30からバッファ70、71を省略すると共に、第1実施例の制御メモリ60と若干異なる制御メモリ60'を具備した構成である。

【0075】この制御メモリ60'は、マイクロプロセッサ50のプログラムや制御データを格納する半導体メモリであり、ダンプ済みデータ情報保持テーブル700'と、コピーポインタ820とを備える。ダンプ済みデータ情報保持テーブル700'は、通常のダンプ順よりも優先してダンプを行ったデータブロックのLBAを保持するテーブルである。コピーポインタ820は、ダンプを行っていない未ダンプ領域の先頭ブロックのLBAを指すポインタである。

【0076】図9、図10および図11は、上記ダンプ機能つきディスク装置30'のダンプ処理、COPYコマンド処理およびWRITEコマンド処理のフローチャートである。また、図12は、上記ダンプ機能つきディスク装置30'の動作を模式的に示す概念図である。

【0077】図9～図12より理解されるように、このダンプ機能つきディスク装置30'では、ダンプ開始からダンプ終了までの間に、未ダンプ領域のデータに対するデータ更新要求を受けたときは、優先的に当該データをダンプ先のダンプ機能なしディスク装置31へ転送し、その後で前記データを更新する。すなわち、ダンプ中にCPU11から未ダンプ領域のデータに対するWRITEコマンドを受けると、優先的に磁気ディスク100から更新前データを読み出してダンプ機能なしディスク装置31へ転送する。そして、ライト対象データを磁気ディスク100に書き込み、CPU11へWRITEコマンドの完了を通知する。また、優先的にダンプしたデータブロックのLBAをダンプ済みデータ情報保持テーブル700'に登録しておき、通常の順でのダンプ処理をスキップさせる。

【0078】この第2実施例のダンプ機能つきディスク装置30'によれば、WRITEコマンドに対する応答時間は長くなるが、バッファ70、71が不要になり、構成が簡単になる。さらに、WRITEコマンドが完了した後は、更新前と更新後のデータがいずれもディスク装置に格納されるため、停電などのためにデータが失われることはない。従って、安価で且つ高信頼性のシステムが得られる。

【0079】（実施例3）図13は、本発明の第3実施例による記憶装置を含む計算機システムの要部構成図である。

【0080】CPU10とCPU11は、SCSIバス20を介して、本発明の第3実施例のダンプ機能つきディスク装置30と、磁気テープ装置900とに接続される。磁気テープ装置900は、データブロックごとのアクセスをサポートするものとする。

【0081】ダンプ機能つきディスク装置30は、上記第1実施例のダンプ機能つきディスク装置30とほぼ同じ構成であるが、ダンプするデータブロックにダンプ元のLBAを付加する「拡張COPYコマンド」をサポートする点と、ダンプ先のデータブロックをダンプ元に復元する「RESTOREコマンド」をサポートする点

が異なっている。

【0082】次に、ダンプ処理の手順を説明する。まず、CPU10は、ダンプ機能つきディスク装置30に「拡張COPYコマンド」を発行して、ダンプ元の磁気ディスク（図示せず）のデータブロックをダンプ先の磁気テープ装置900にダンプするように指示する。

【0083】ダンプ機能つきディスク装置30は、図3と同様のダンプ処理を実行する（図3の「ダンプ機能なしディスク装置31」を「磁気テープ装置900」に置換し、「ダンプ機能つきディスク装置30」を「ダンプ機能つきディスク装置30」に置換し、「COPYコマンド」を「拡張COPYコマンド」に置換し、「COPYコマンド処理」を「拡張COPYコマンド処理」に置換する）。

【0084】拡張COPYコマンド処理は、図4と同様である（図4のステップ340、350で、データブロックにダンプ元のLBAを付加したLBA付加型データブロックをダンプ先へWRITEする）。

【0085】図14に、LBA付加型データブロックの一例を示す。LBA付加型データブロック1110（例えば、長さ516バイト）は、ダンプ元のLBA1102と、データブロック1103（例えば、データ長512バイト）とを組み合わせた構造である。

【0086】拡張COPYコマンド処理中に受けたWRITEコマンドに対するWRITEコマンド処理は、図5と同様である（図5の「COPYコマンド処理」を「拡張COPYコマンド処理」に置換する）。

【0087】フラッシュ処理は、図6と同様である（図6のステップ610で、LBA付加型データブロックを、当該データブロックの転送の直前に他のデータブロックが転送されていた記憶領域の次の記憶領域にWRITEする）。

【0088】図15は、上記ダンプ機能つきディスク装置30の動作を模式的に示す概念図である。制御メモリ60のコピーポインタ820を、ダンプ元の磁気ディスク100の先頭LBAから1ブロックずつシーケンシ

ャルに移動させ、ポイントされたデータブロックにLBAを付加したLBA付加型データブロックをダンプ先の磁気テープ900aに順に書き込む。図示の斜線部分は、ダンプ済の領域を示す。

【0089】ダンプ開始からダンプ終了までの間（拡張COPYコマンド処理中）に、「WRITEコマンド」を受けると、更新前データブロック1010を更新前データブロック保持用領域110の領域1011に退避する。領域1011のアドレスは更新前データ情報保持テーブル700に格納しておく。バッファ71についても同じことを2重に行う。

【0090】更新前データブロック1010が書き込まれていた領域Paまでコピーポインタ820が来た契機か又は更新前データ保持用領域110が“フル”になった契機のいずれか早い方の契機で、退避していた更新前データブロック1010にLBAを付加したLBA付加型データブロックを磁気テープ900aの領域1012に書き込む。領域1012は、直前の書き込み領域に続く領域であり、更新前データブロック1010のLBAとは無関係になってしまう。このため、LBAを付加している。

【0091】図16は、ダンプ機能つきディスク装置30によるRESTOREコマンド処理のフローチャートである。ステップ1200では、CPU10から「RESTOREコマンド」を受領し、直ちにCPU10に「DISCONNECT」を発行し、SCSIバス20との接続を切断する。

【0092】ステップ1210では、磁気テープ装置900に「LOCATEコマンド」を発行し、磁気テープ900aを先頭に位置づける。ステップ1220では、磁気テープ装置900に「READコマンド」を発行し、1つのLBA付加型データブロックをバッファ70（バッファ71でもよい）に読み込む。

【0093】ステップ1230では、磁気ディスク100に「WRITEコマンド」を発行し、LBA付加型データブロック1100のデータブロックだけを、当該LBAの領域に書き込む。これにより、磁気テープ900aに書き込まれたデータブロックがLBAの順序どおりに磁気テープ900a上で並んでいなくとも、磁気ディスク100には元のLBAに対応した領域（位置）にデータブロックを復元できる。

【0094】ステップ1240では、磁気テープ900aに書き込まれたLBA付加型データブロックをすべて読み終えたか否か判定する。読み終えていればステップ1250に進み、読み終えていなければ前記ステップ1220に戻る。ステップ1250では、CPU10に「RESELECT」を発行し、ステータスとコマンドコンプリートメッセージを送り、RESTOREコマンドを終了する。

【0095】上記第3実施例のダンプ機能つきディスク

装置30"によれば、データブロックにダンプ元のLBAを付加したLBA付加型データブロックを磁気テープ900aにダンプするので、ダンプ元のLBAと磁気テープ900aの書き込み領域とが無関係であっても、データブロックの元の領域(位置)の情報を欠落させることなく、ダンプ開始時点でのダンプ対象のデータブロックを完全にダンプすることが出来る。また、ダンプ先のLBA付加型データブロックを用いて、元のLBAに対応した領域にデータブロックを復元するので、ダンプ開始時点でのダンプ対象のデータブロックを完全に復元することが出来る。

【0096】上記第3実施例は、次のように変形してもよい。

①LBA付加型データブロックに、さらに当該データブロックの「更新時刻」や「ダンプ時刻」などの時刻識別子を付加すれば、同じLBAのデータブロックが複数あっても、どれがダンプ開始時点のデータブロックかを特定できるので、フラッシュ処理で既にダンプ済みとされた領域についても、LBA順のダンプ処理で再びダンプを行なってもよい。

【0097】②LBA付加型データブロックに、さらに当該データブロックの「更新時刻」や「ダンプ時刻」などの時刻識別子を付加すれば、同じLBAのデータブロックが複数あっても、どれがダンプ開始時点のデータブロックかを特定できるので、データ更新済み領域をダンプする順が来た時に、バッファに退避していた更新前データブロックを無視し、磁気ディスクの更新後データブロックをダンプ先に転送し、その後、適当な契機(例えばフラッシュ処理)で、バッファに退避していた更新前データブロックをダンプ先に転送してもよい。

【0098】上記①②によれば、拡張COPYコマンド処理とフラッシュ処理とを全く独立に(2つの処理で共用するバッファなどのデータ構造を持たずに)動作させることが出来る。

③その他、第1実施例と同様の変形が可能である。

【0099】(実施例4)図17は、本発明の第4実施例の記憶装置を含む計算機システムの要部構成図である。

【0100】本発明の第4実施例の記憶装置であるダンプ機能つきディスク装置30'''の構成は、基本的に第2実施例と同様な構成であるが、第2実施例のダンプ機能つきディスク装置30'の制御メモリ60'と若干異なる制御メモリ60''を具備した構成である。

【0101】この制御メモリ60''は、マイクロプロセッサ50のプログラムや制御データを格納する半導体メモリであり、コピーポインタ820を備える。第2実施例のダンプ機能つきディスク装置30'の制御メモリ60'とは異なり、ダンプ済みデータ情報保持テーブル700'は備えない。

【0102】テープ装置2000は、SCSIインタフ

ェース制御部40、マイクロプロセッサ50、制御メモリ2030、デバイスインタフェース制御部90、磁気テープ2010と2020を有している。制御メモリ2030は、マイクロプロセッサ50のプログラムや制御データを格納する半導体メモリである。

【0103】図18、図19および図20は、上記ダンプ機能つきディスク装置30'''のダンプ処理、COPYコマンド処理およびWRITEコマンド処理のフローチャートである。

【0104】図18~図20により理解されるように、このダンプ機能つきディスク装置30'''では、ダンプ開始からダンプ終了までの間に、未ダンプ領域のデータに対するデータ更新要求を受けたときは、当該データをダンプ先の磁気テープ2020へ優先的に転送し、その後、前記データを更新する。すなわち、ダンプ中にCPU11から未ダンプ領域のデータに対するWRITEコマンドを受けると、磁気ディスク100から更新前データを優先的に読み出して、当該データのLBAを付加したうえで磁気テープ2020へ転送し、格納する。そして、ライト対象データを磁気ディスク100に書き込み、CPU11へWRITEコマンドの完了を通知する。

【0105】一方、コピーポインタ820に従って、シーケンシャルに行われるCOPYコマンド処理では、磁気ディスク100のデータは前記更新前データとは異なる磁気テープ2010に格納される。データには、LBAなどの識別子は付加されない。COPYコマンド処理235''では、対象とするデータが、ダンプ開始以来、更新されていた場合には、更新後のデータを磁気テープ2010に格納する。このため、ダンプ開始時の磁気ディスク100の内容を回復するためには、磁気テープ2010の内容に、磁気テープ2020に待避されている更新前データを反映させなければならない。これを行うのが、図21に示すRESTOREコマンド処理である。

【0106】ステップ1200では、CPU10から「RESTOREコマンド」を受け取って、直ちにCPU10に「DISCONNECT」を発行し、SCSIバス20との接続を切断する。ステップ2100では、テープ装置2000に「LOCATEコマンド」を発行し、磁気テープ2010を先頭に位置づける。

【0107】ステップ2110では、テープ装置2000に「READコマンド」を発行し、磁気テープ2010から1つのブロックを読み込む。ステップ2120では、磁気ディスク100に「WRITEコマンド」を発行し、ブロックを書き込む。

【0108】ステップ2130では、磁気テープ2010に書き込まれたブロックをすべて読み終えたか否かを判定する。読み終えていればステップ2140に進み、読み終えていなければ前記ステップ2110に戻る。ス

ステップ2140では、テープ装置2000に「LOCATEコマンド」を発行し、磁気テープ2020を最終ブロックに位置づける。

【0109】ステップ2150では、テープ装置2000に「READ REVERSEコマンド」を発行し、磁気テープ2020から、媒体の後ろから前方向に向かって、1つのLBA付加型データブロックを読む。ステップ2160では、READ REVERSEコマンドで読まれたデータのバイト順を逆転させる。これは、READ REVERSEコマンドが、ブロックを逆順に読むばかりでなく、ブロック内のバイト順も逆になったデータを返すからである。

【0110】ステップ2170では、磁気ディスク100に「WRITEコマンド」を発行し、LBA付加型データブロック1100のデータブロックだけを、当該LBAの領域に書き込む。ステップ2180では、磁気テープ2020に書き込まれたLBA付加型データブロックをすべて読み終えたか否か判定する。読み終えていればステップ1250に進み、読み終えていなければ前記ステップ2150に戻る。

【0111】ステップ1250では、CPU10に「RESELECT」を発行し、ステータスとコマンドコンプリートメッセージを送り、RESTOREコマンドを終了する。

【0112】ステップ2150で、磁気テープ2020のブロックを逆順に読む理由を以下に示す。ダンプ処理の間に同一ブロックに複数回更新があった場合、磁気テープ2020にも、更新前データが複数個格納される。これらデータは、磁気テープ2020に発生順に格納されるために、もっとも小さいブロック番号を持つものが、ダンプ処理開始時のデータを持つものであり、それ以外のブロックはRESTORE時には無視すべきである。磁気テープ2020のブロックを逆順に読み、磁気ディスク100に反映させることによって、同一ブロックの更新前データが複数個あっても、最終的にはもっとも小さいブロック番号を持つ更新前データが有効となる。

【0113】上記第4実施例のダンプ機能つきディスク装置30'''は、ダンプデータを書き込むための磁気テープと、更新前データを書き込むための磁気テープを別にしている。さらに、更新前データにはLBAを付加して格納し、RESTORE時には格納時と逆順にブロックを読んで、ダンプデータを書き込んだ磁気テープのデータに反映させる。このため、前述の第1から第3実施例のように更新前データ保持テーブルがなくとも、ダンプ開始時点でのダンプ対象のデータブロックを完全にダンプし、また復元することが出来る。

【0114】さらに、ダンプデータを書き込むための磁気テープに格納されるデータブロックには、LBAなどの余分な識別子が不要であるため、テープの容量が無駄

にならない。また、本実施例においては、ダンプ先の記憶媒体へのアクセスは、完全にシーケンシャルであるため、テープなど、ランダムアクセスに適さない記憶媒体を使用することができる。ディスクなどランダムアクセスに適した記憶媒体であっても、一般的にはシーケンシャルアクセスのほうが、ランダムアクセスに比べて高性能であるため、ダンプ処理の時間を短縮できる。

【0115】上記第1、第2、第3、および第4実施例において、CPUとディスク装置ないしテープ装置との接続は、SCSIバスによって行われる。これを、ESCON(Enterprise Systems Connection)ディレクタによって行う構成も可能である。

【0116】図22は、図1に示す計算機システムのSCSIバスを、ESCONディレクタに置き換えたものであり、第1の実施例に示した動作に類似の動作をする。図23は、図8に示す計算機システムのSCSIバスを、ESCONディレクタに置き換えたものであり、第2の実施例に示した動作に類似の動作をする。

【0117】図24は、図13に示す計算機システムのSCSIバスを、ESCONディレクタに置き換えたものであり、第3の実施例に示した動作に類似の動作をする。図25は、図17に示す計算機システムのSCSIバスを、ESCONディレクタに置き換えたものであり、第4の実施例に示した動作に類似の動作をする。

【0118】

【発明の効果】本発明の記憶装置によれば、CPUを介在させずにダンプ処理を行うことが出来ると共に、ダンプ開始からダンプ終了までの間にデータ更新を行なっても、ダンプ開始時点でのダンプ対象データを完全にダンプすることが出来る。このため、記憶装置としての可用性を向上できる。

【図面の簡単な説明】

【図1】本発明の第1実施例の記憶装置を含む計算機システムの要部構成図である。

【図2】本発明の第1実施例の更新前データ情報保持テーブルの構成図である。

【図3】本発明の第1実施例のダンプ処理の手順を示すフローチャートである。

【図4】本発明の第1実施例のCOPYコマンド処理の手順を示すフローチャートである。

【図5】本発明の第1実施例のWRITEコマンド処理の手順を示すフローチャートである。

【図6】本発明の第1実施例のフラッシュ処理の手順を示すフローチャートである。

【図7】本発明の第1実施例のダンプ処理の概念を説明するデータフロー図である。

【図8】本発明の第2実施例の記憶装置を含む計算機システムの要部構成図である。

【図9】本発明の第2実施例のダンプ処理の手順を示すフローチャートである。

21

【図10】本発明の第2実施例のCOPYコマンド処理の手順を示すフローチャートである。

【図11】本発明の第2実施例のWRITEコマンド処理の手順を示すフローチャートである。

【図12】本発明の第2実施例のダンプ処理の概念を説明するデータフロー図である。

【図13】本発明の第3実施例の記憶装置を含む計算機システムの要部構成図である。

【図14】LBA付加型データブロックのデータ構造図である。

【図15】本発明の第3実施例のダンプ処理の概念を説明するデータフロー図である。

【図16】本発明の第3実施例のRESTOREコマンド処理の手順を示すフローチャートである。

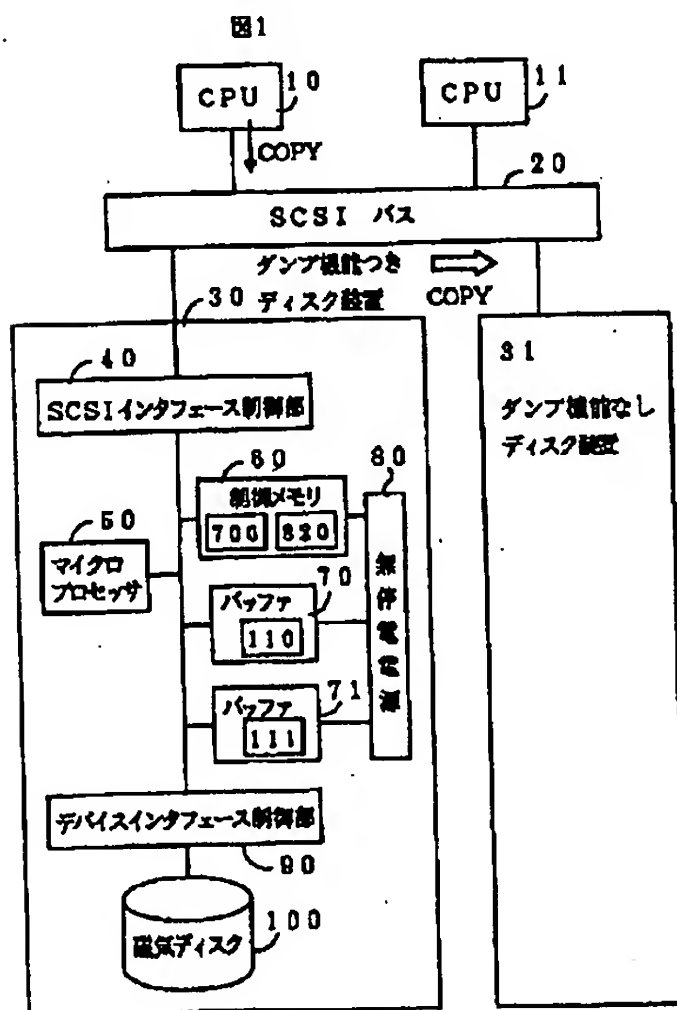
【図17】本発明の第4実施例の記憶装置を含む計算機システムの要部構成図である。

【図18】本発明の第4実施例のダンプ処理の手順を示すフローチャートである。

【図19】本発明の第4実施例のCOPYコマンド処理の手順を示すフローチャートである。

【図20】本発明の第4実施例のWRITEコマンド処理の手順を示すフローチャートである。

【図1】



22

【図21】本発明の第4実施例のRESTOREコマンド処理の手順を示すフローチャートである。

【図22】本発明の記憶装置を含む計算機システムの要部構成図である。

【図23】本発明の記憶装置を含む計算機システムの要部構成図である。

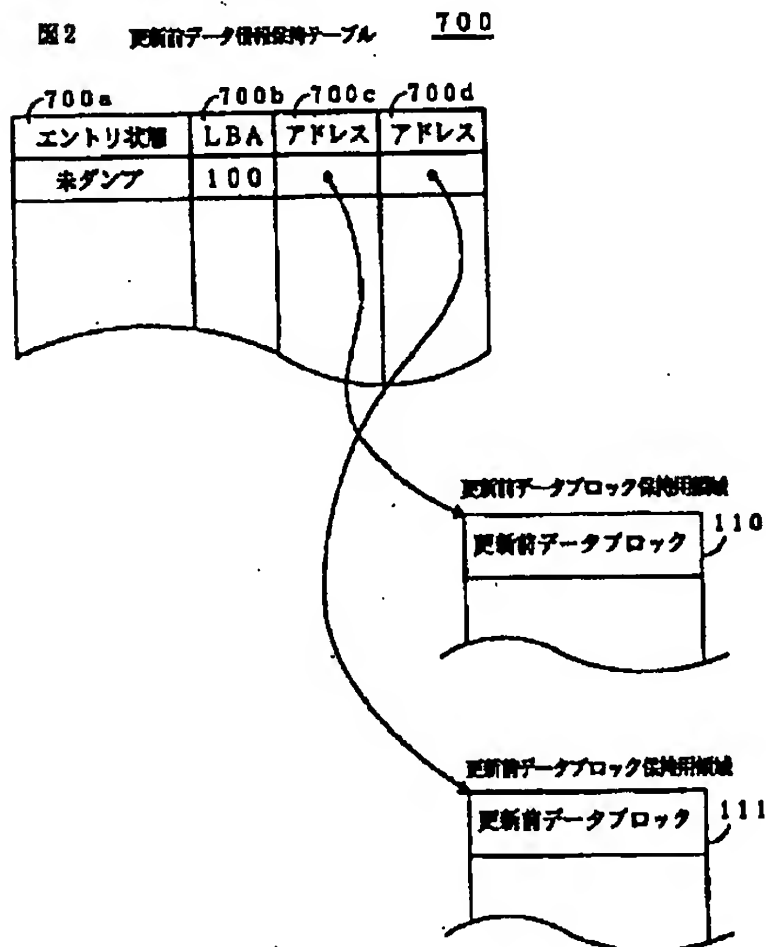
【図24】本発明の記憶装置を含む計算機システムの要部構成図である。

【図25】本発明の記憶装置を含む計算機システムの要部構成図である。

【符号の説明】

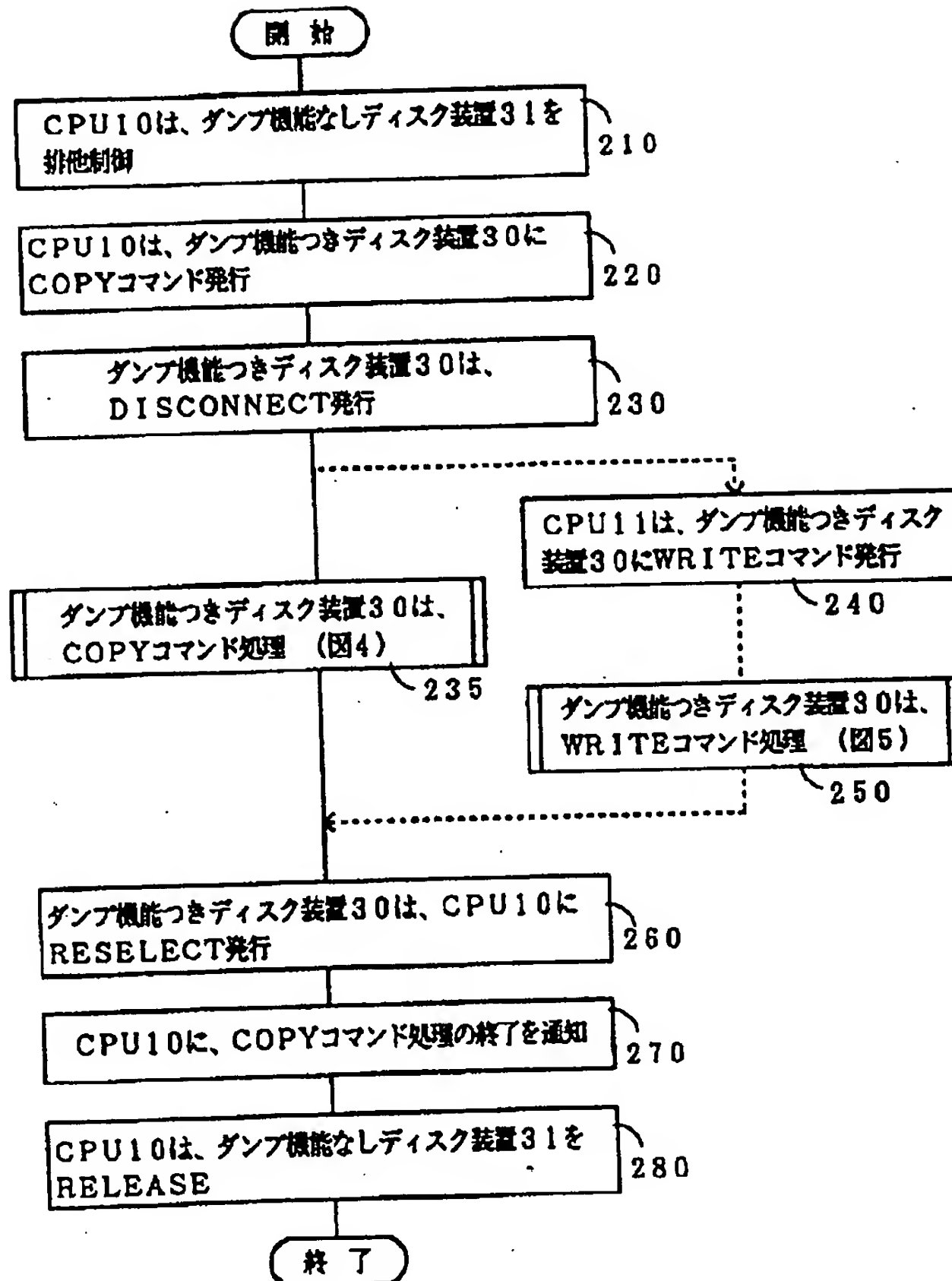
10, 11: CPU、 20: SCSIバス、 30, 30', 30'', 30''' : ダンプ機能つきディスク装置、 31: ダンプ機能なしディスク装置、 40: SCSIインタフェース制御部、 50: マイクロプロセッサ、 60, 60', 60'' : 制御メモリ、 70, 71: バッファ、 80: 無停電電源、 90: デバイスインタフェース制御部、 100: 磁気ディスク、 110, 111: 更新前データ保持用領域、 700: 更新前データ情報保持テーブル、 700': ダンプ済みデータ情報保持テーブル、 820: コピーポインタ、 900: 磁気テープ装置、 900a: 磁気テープ

【図2】



【図3】

図3

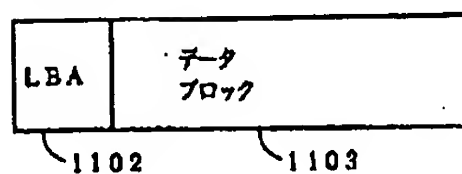


【図14】

図14

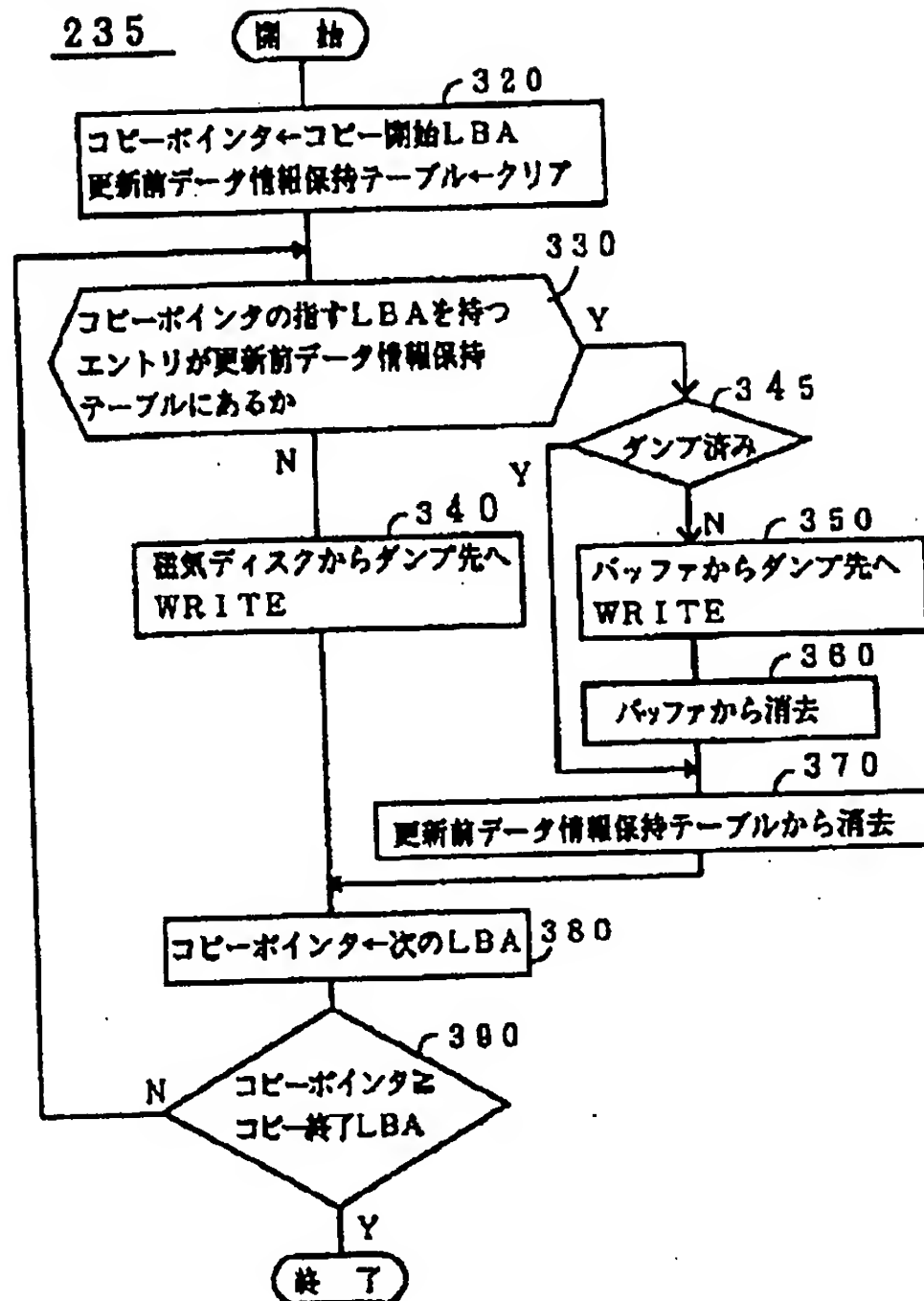
LBA付加型データブロック

1100

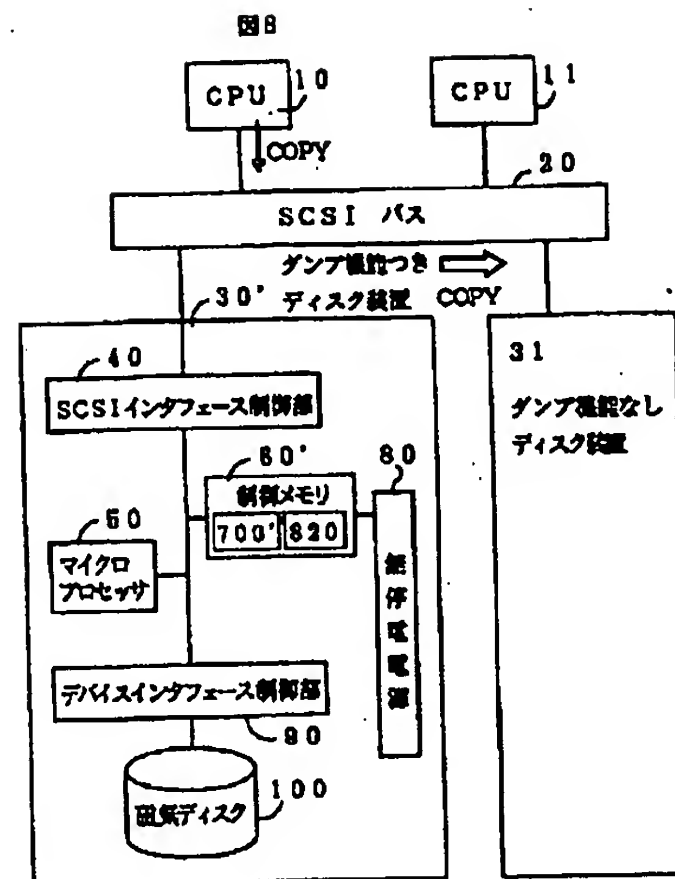


【図4】

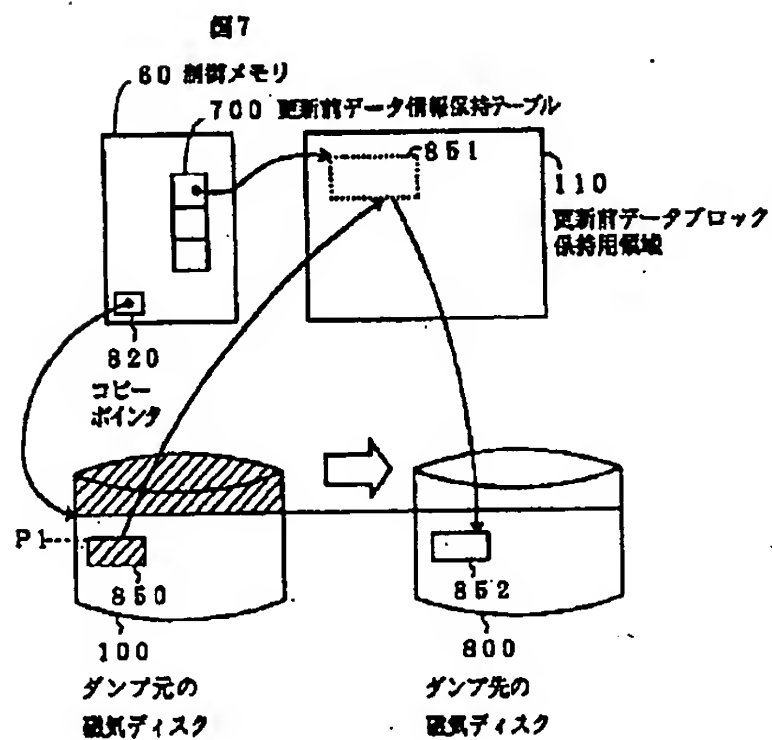
図4



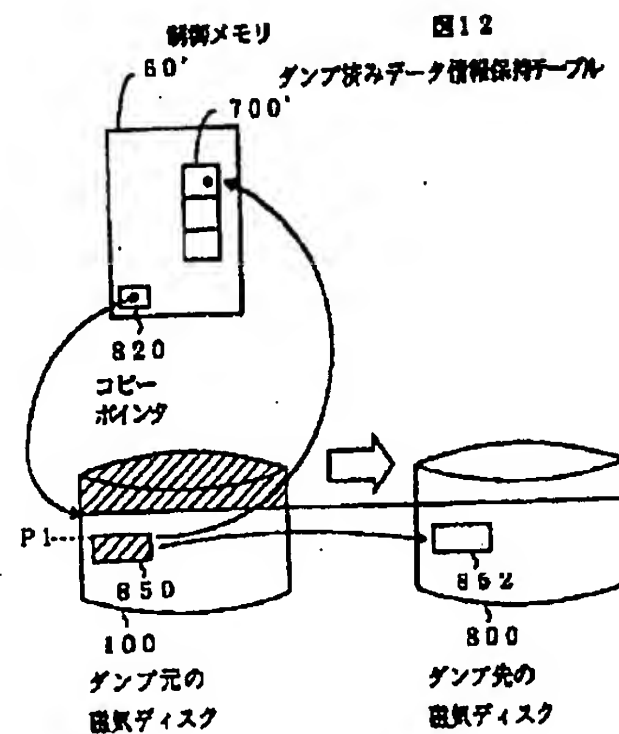
【図8】



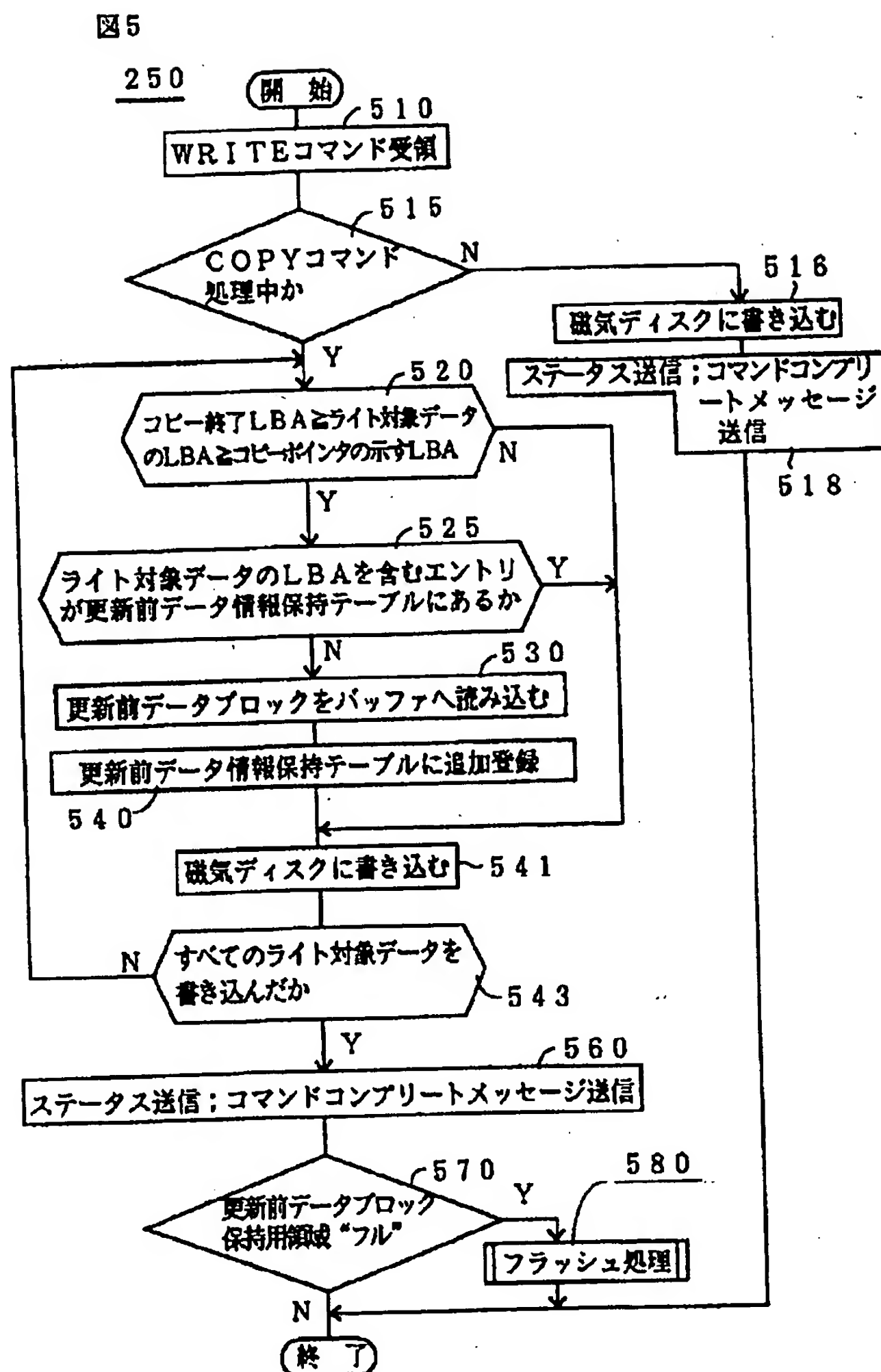
【図7】



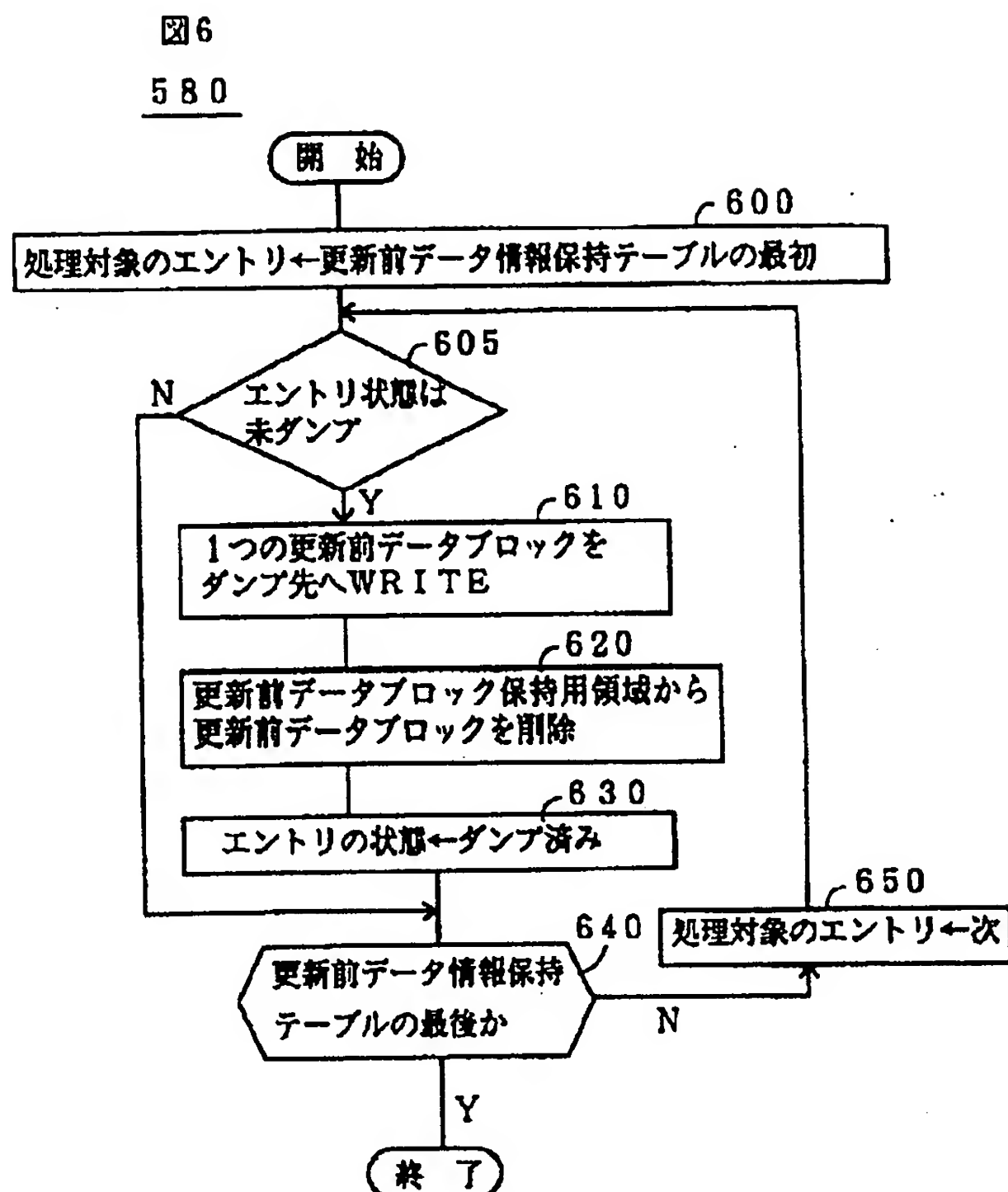
【図12】



【図5】

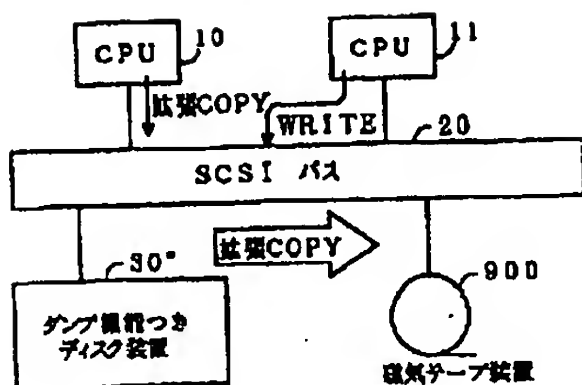


【図6】



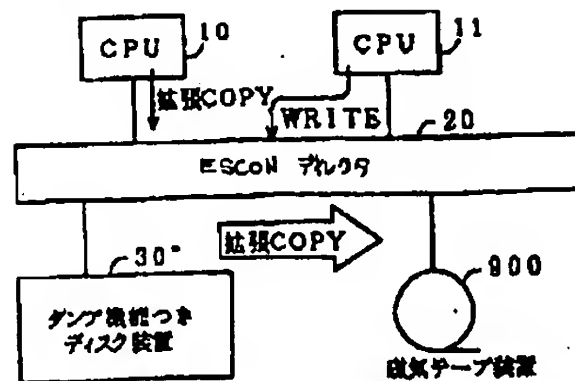
【図13】

図13 計算機システム



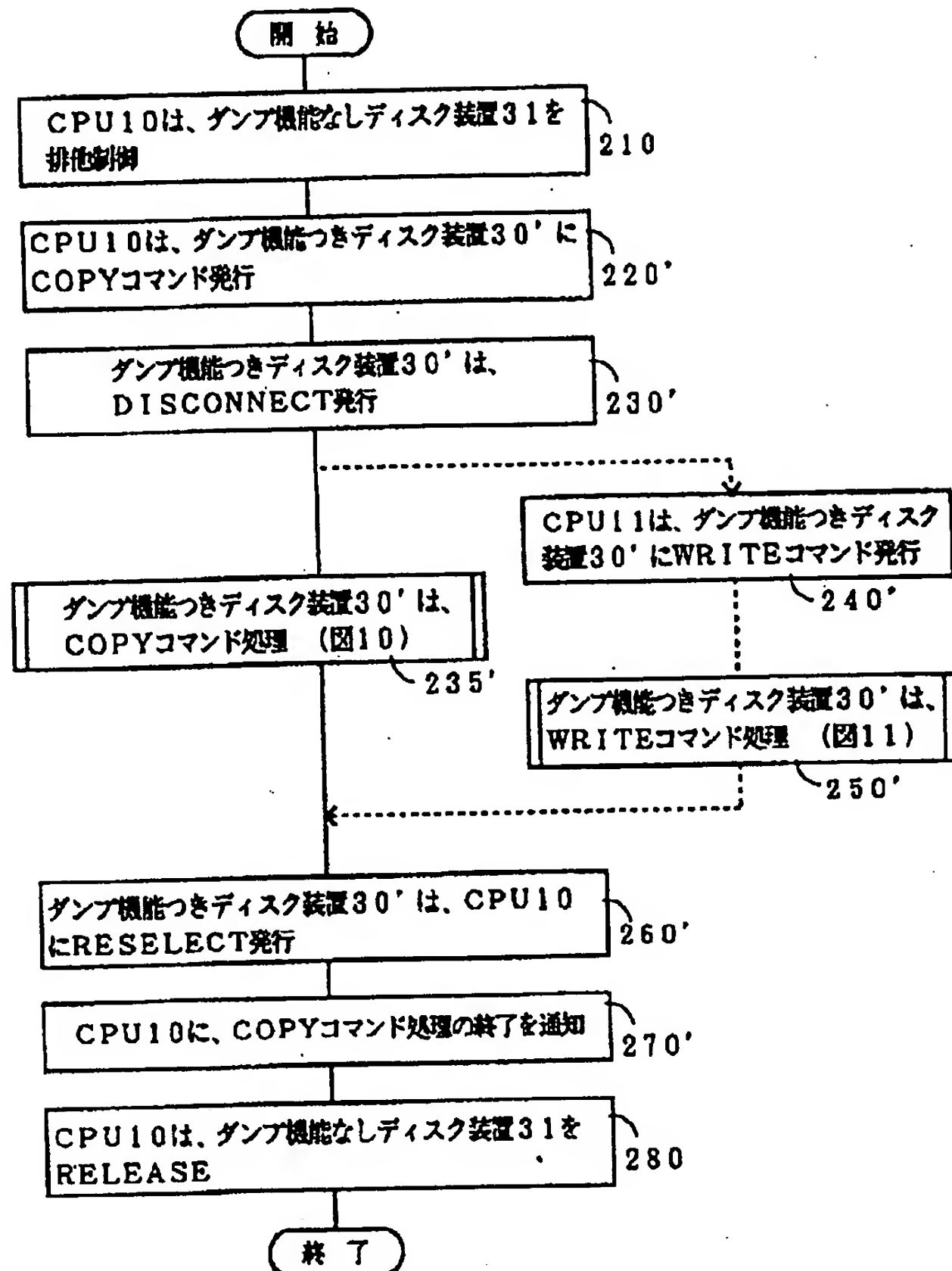
【図24】

図24



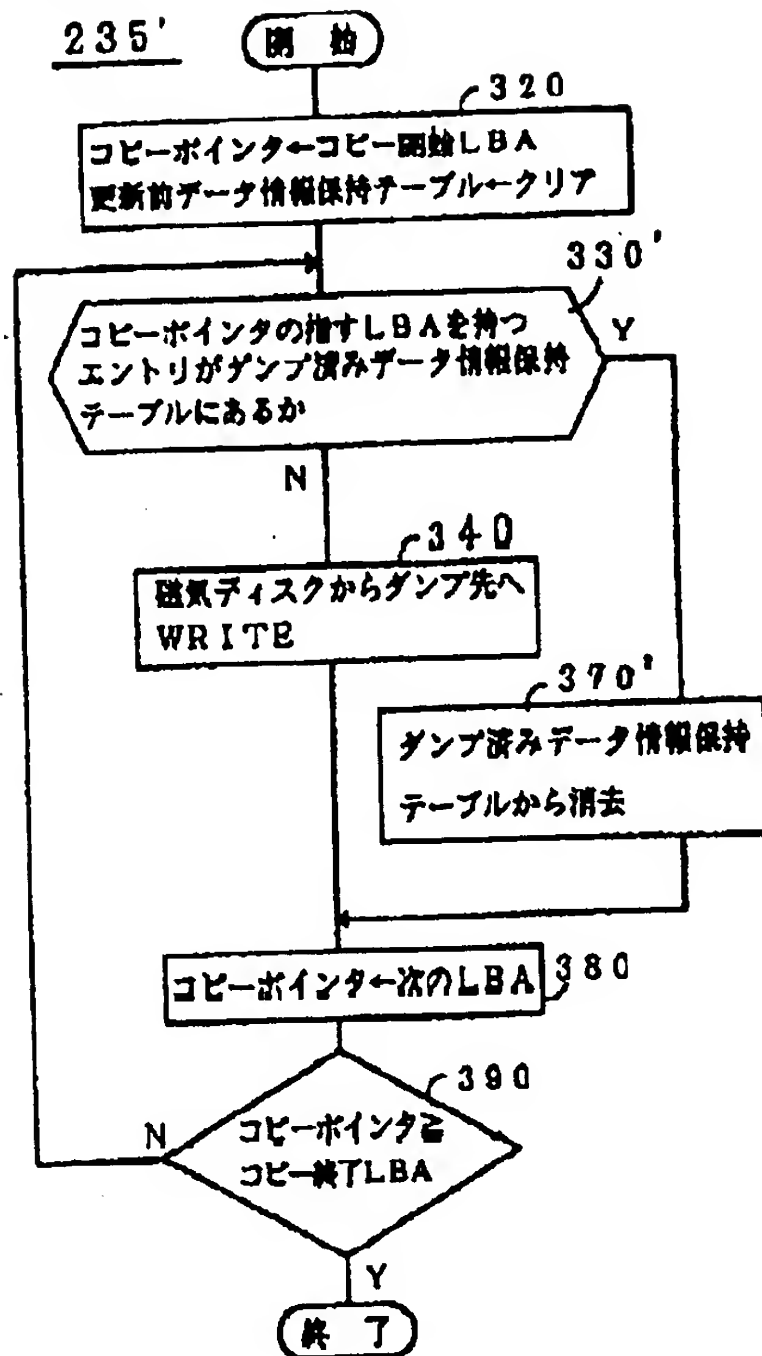
【図9】

図9

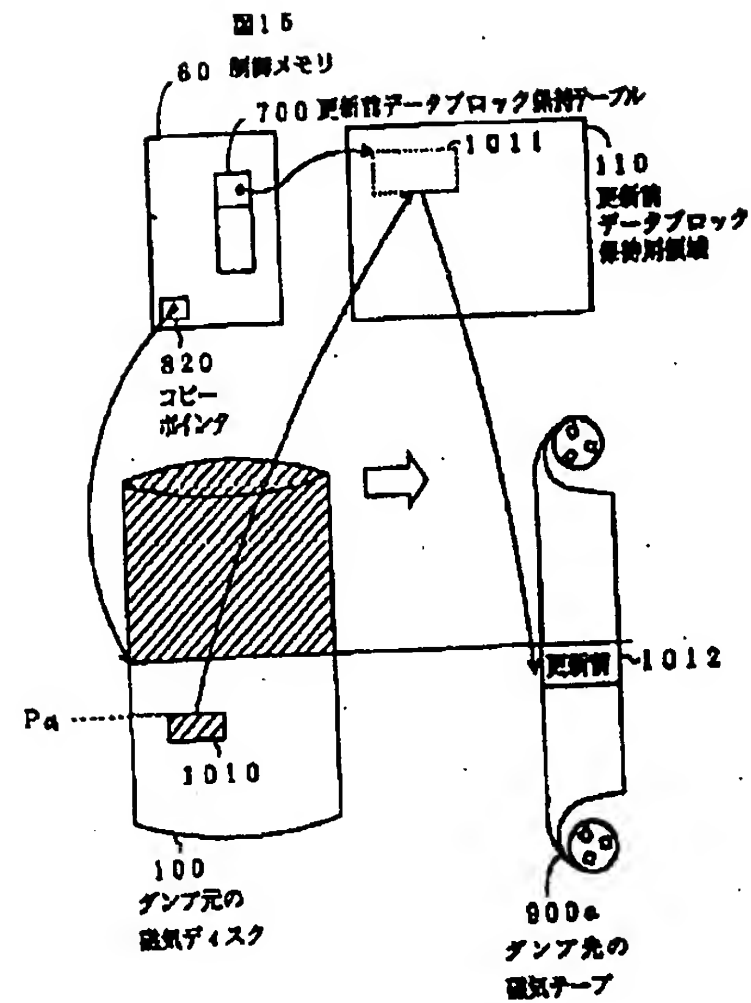


【図10】

図10

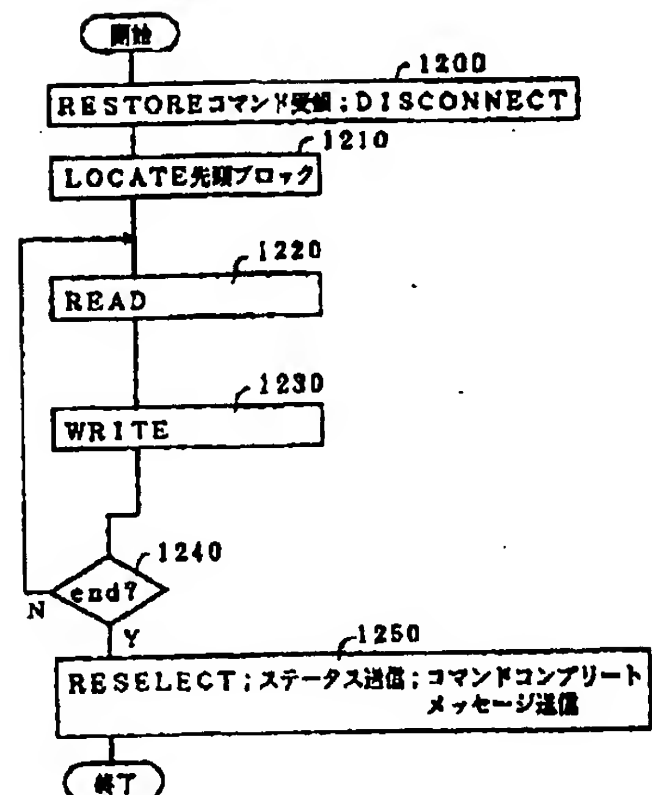


【図15】



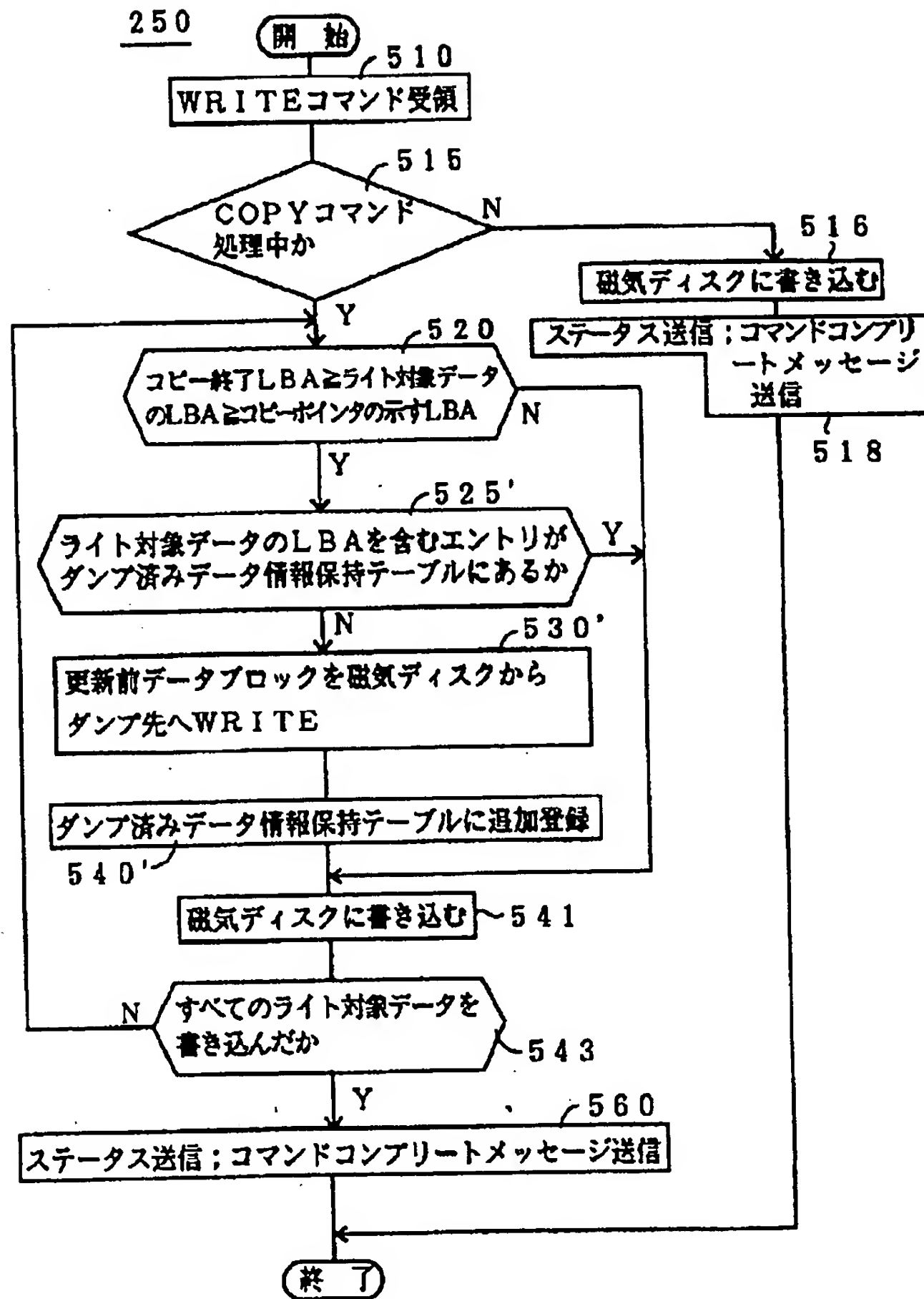
【図16】

図16

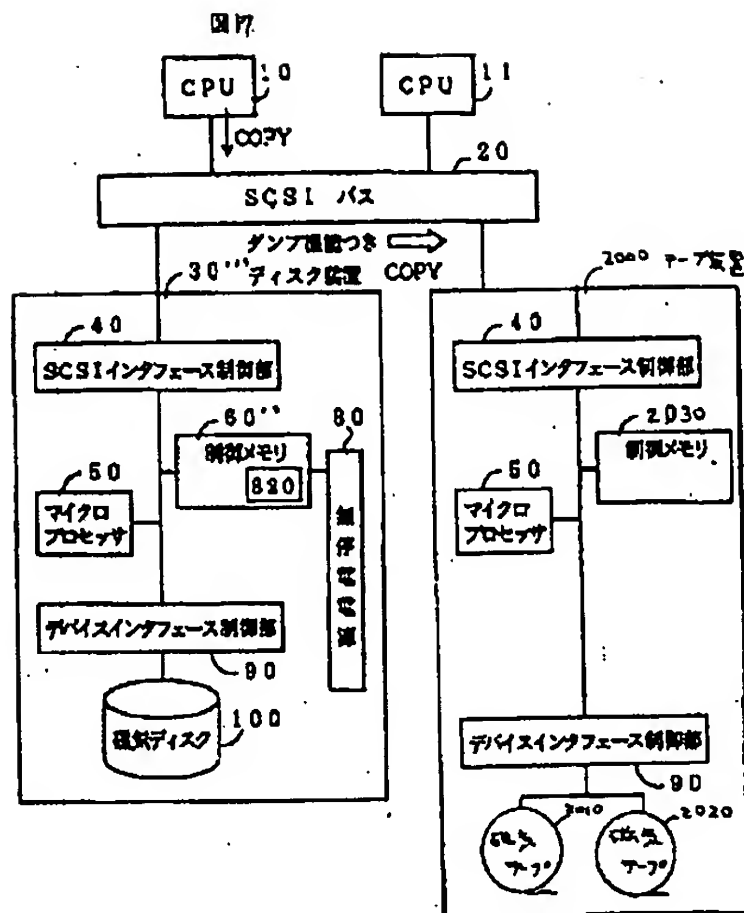


【図11】

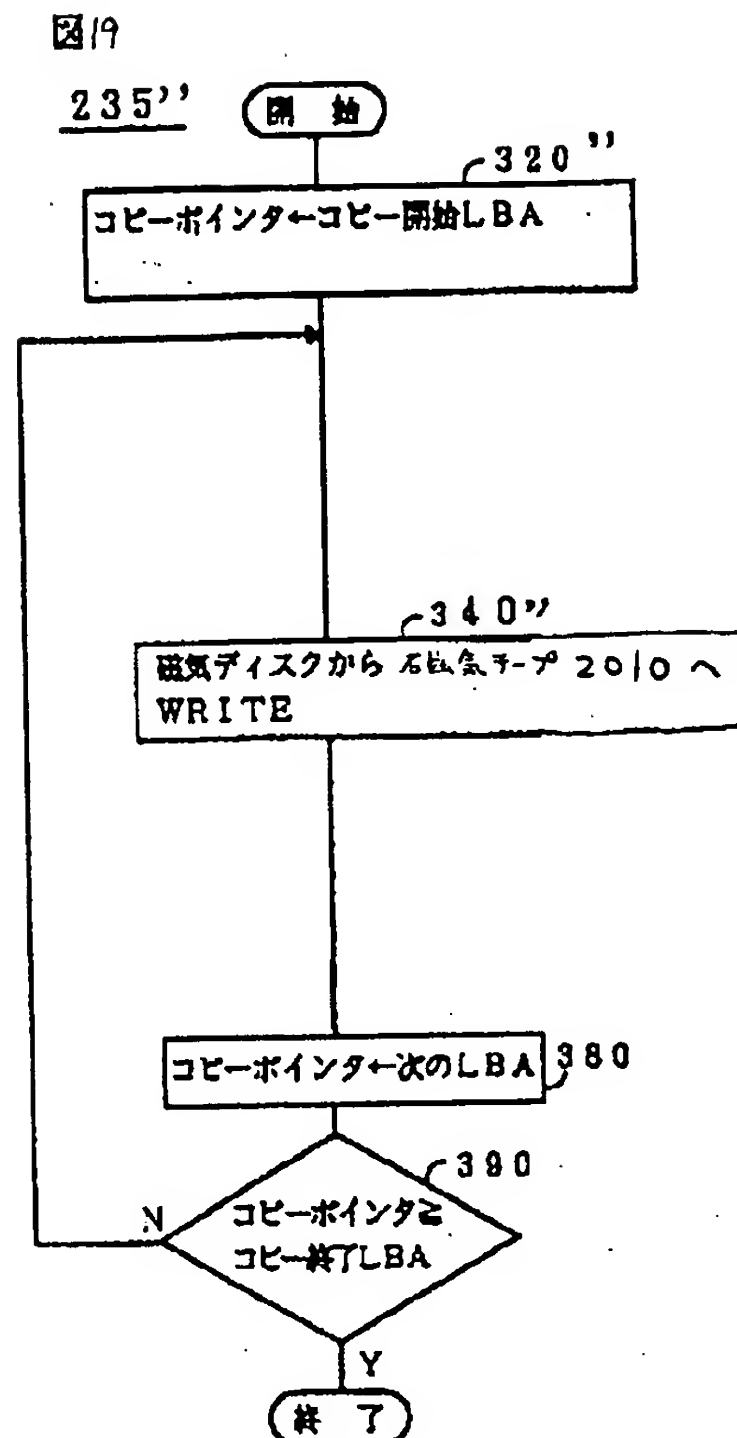
図11



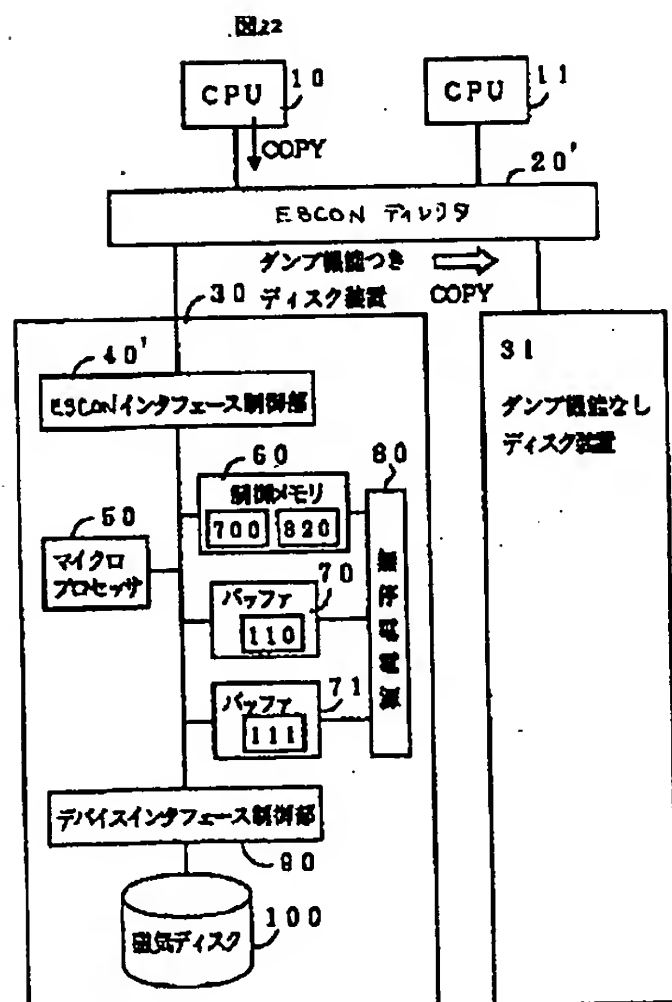
【図17】



【図19】

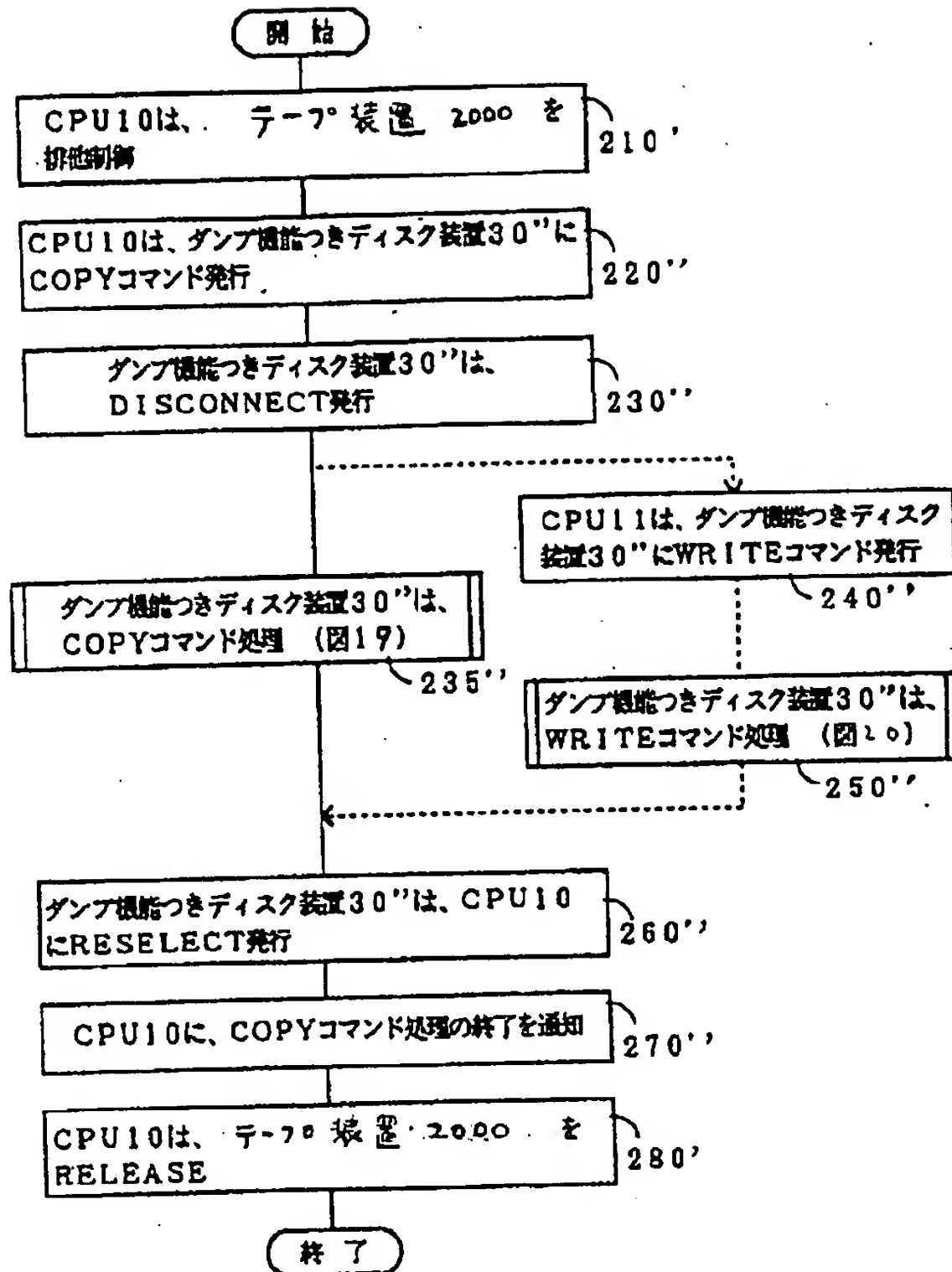


【図22】



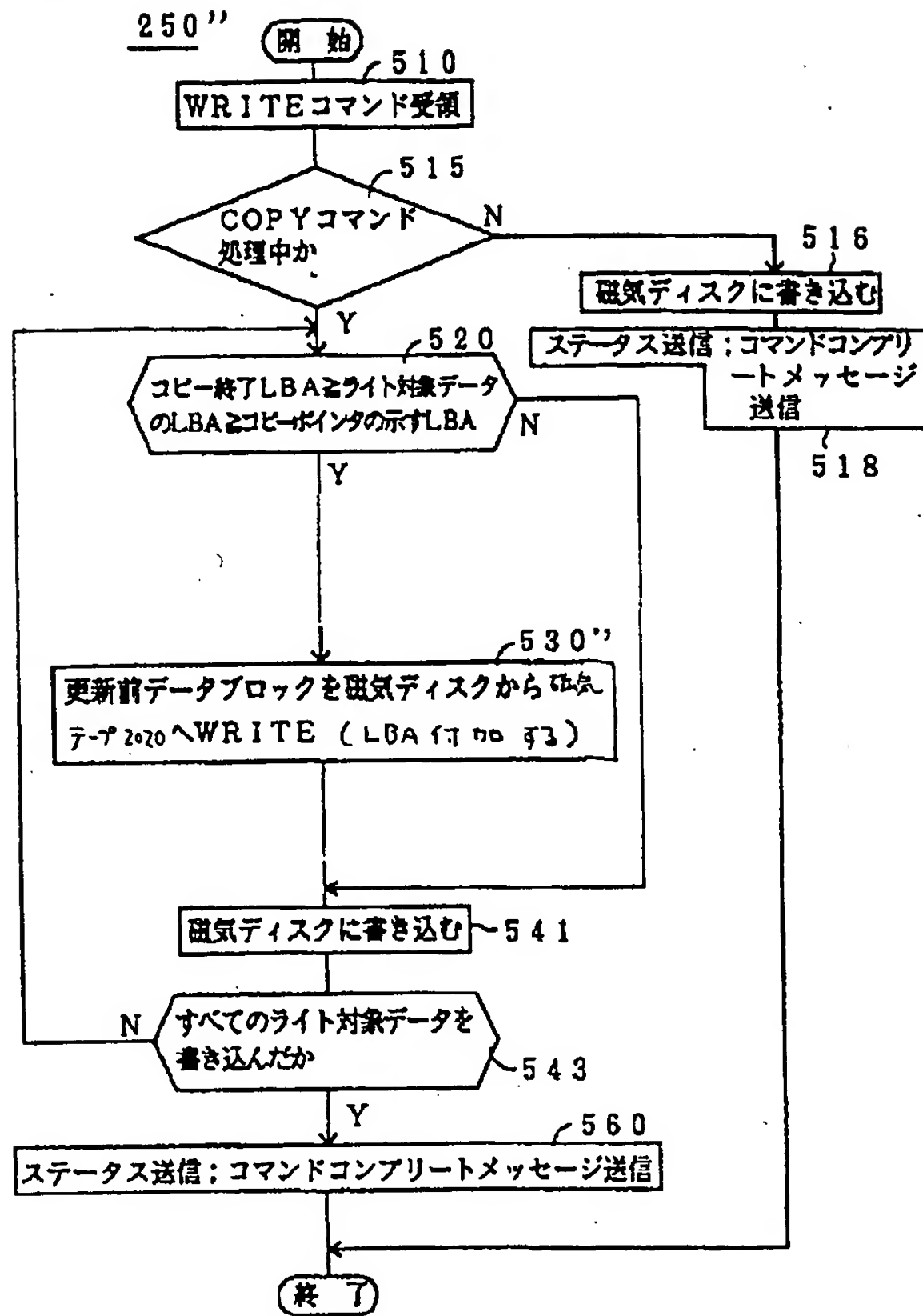
【図18】

図18



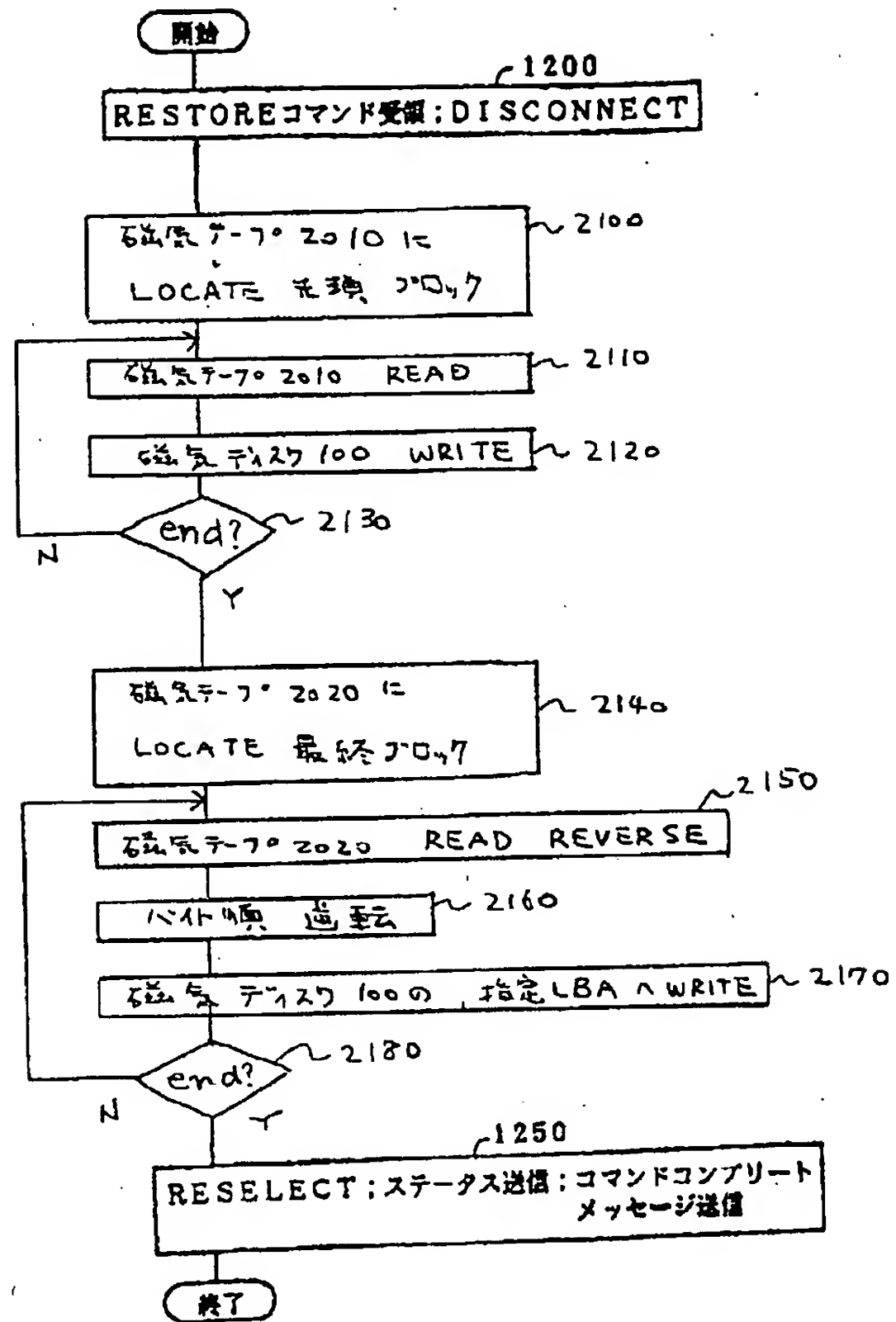
【図20】

図20

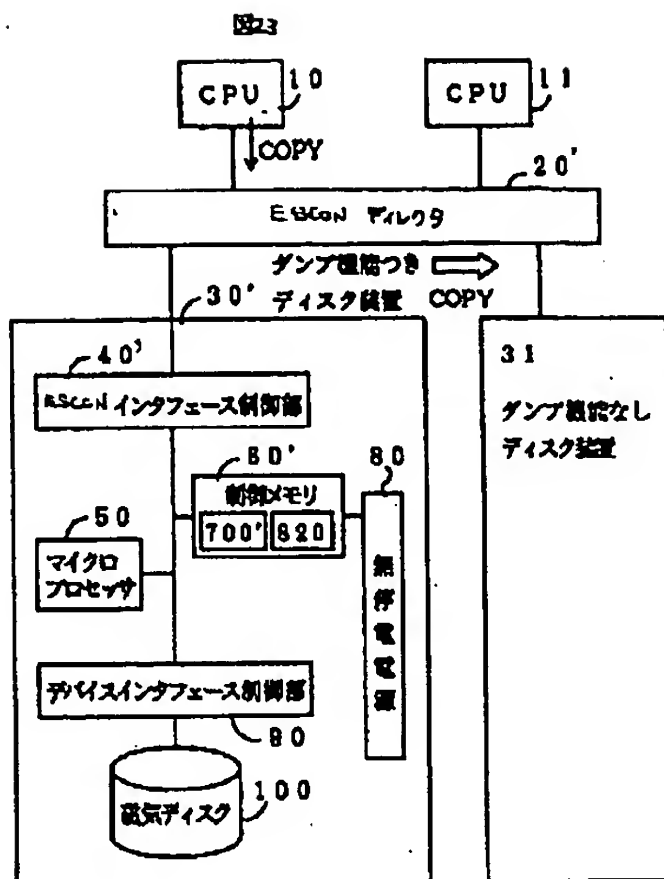


【図21】

図21



【図23】



【図25】

